IGE | IPI
Swiss Federal Institute
of Intellectual Property

New Software Protection Approaches in a World (Co)shaped by AI

Publikation Nr. 13 (2024-10)

Editor
Swiss Federal Institute of Intellectual Property
Stauffacherstrasse 65/59g
CH-3003 Bern

Authors

Peter Georg Picht
Florent Thouvenin
Valerie Brunner
Joël Donzé

CIPCO Universität Zürich
Freiestrasse 15
CH-8032 Zürich
www.cipco.uzh.ch

Download
Swiss Federal Institute of Intellectual Property, Bern
www.ipi.ch

Foreword

“For x=1 to 10; Print ‘Hello World’; Next x;” That’s the first ‘computer program’ I ever wrote, as a teenager. It made the words ‘Hello World’ appear ten times on my screen. Admittedly, my debut program wasn’t very creative. It wouldn’t have qualified for protection under either copyright or patent law. Nowadays, a great deal of resources are often needed to create professional software. But there’s only an incentive to invest in the first place if the funds involved can at least be recouped. That’s why software protection must be as efficient as possible. The current solution involving patent and copyright law is unsatisfactory for many, and the rapidly increasing use of generative artificial intelligence, especially in the area of code generation, also calls the very foundations of these laws into question.

This is therefore a good time to contemplate new forms of software protection that are in line with current technological developments. Hence, we commissioned Florent Thouvenin and Peter Picht, who are both professors in the Faculty of Law at the University of Zurich, to produce a paper on the subject.

The challenge with this kind of proposal is to come up with a new protective method that is more attractive for all stakeholders than the current ones. Today’s methods are set out in international agreements, which would be very difficult to change. This paper therefore suggests creating a sui generis right for software that is based on the current protective systems under copyright and patent law but that is tailored to the needs of effective software protection.

I see this proposal as a key contribution to the burgeoning debate about the future of software protection. I hope that it will be given due consideration and that it will inspire many other researchers and practitioners to think about this topic. We’ll be keeping a close eye on the debate in any case, and we’ll offer the paper a platform at various events.

I’d like to thank all the authors involved for their valuable contributions. They definitely show young people that it makes sense to acquire a better knowledge of the art of programming than I did in my youth, not least thanks to appropriate IP protection.

Catherine Chammartin
Director General of the Swiss Federal Institute of Intellectual Property

Bern, October 2024

Vorwort

«For x = 1 to 10; Print 'Hello World'; Next x;» So sah in meiner Jugend das erste «Computer-Programm» aus, das ich je geschrieben habe. Es liess zehn Mal «Hello World» auf dem Bildschirm erscheinen. Zugegeben: Sehr kreativ war mein damaliges Erstlingswerk nicht. Es hätte sich weder für einen urheberrechtlichen noch einen patentrechtlichen Schutz qualifiziert. Für die Erstellung professioneller Software sind heute oft sehr viele Ressourcen notwendig. Nur wenn es eine Möglichkeit gibt, diese Mittel zumindest wieder einzuspielen, besteht ein Anreiz, die Investitionen auch zu tätigen. Daher ist ein möglichst effizienter Softwareschutz notwendig. Die heutige Lösung mittels Patent- und Urheberrecht ist für viele unbefriedigend, und der schnell voranschreitende Einsatz generativer künstlicher Intelligenz gerade auch im Bereich der Code-Generierung stellt ihre Grundfesten noch zusätzlich infrage.

Es ist somit ein guter Zeitpunkt, sich Gedanken um neue, der aktuellen technologischen Entwicklung angepasstere Formen des Softwareschutzes zu machen. Daher haben wir bei Florent Thouvenin und Peter Picht, beides Professoren an der juristischen Fakultät der Universität Zürich, eine entsprechende Studie in Auftrag gegeben.

Die Herausforderung bei einem solchen Vorschlag ist, dass die neue Schutzmethode für alle Beteiligten attraktiver sein muss als die bisherigen Optionen, da letztere in nur sehr schwer veränderbaren internationalen Abkommen festgeschrieben sind. Der vorliegende Beitrag sieht daher vor, für Software ein Sui-Generis-Recht zu schaffen, das zwar auf den bestehenden Schutzsystemen des Patent- und Urheberrechts aufbaut, aber an die Bedürfnisse eines wirksamen Softwareschutzes angepasst ist.

Ich verstehe diesen Vorschlag als einen gewichtigen Beitrag in der aufkeimenden Diskussion um die Zukunft des Softwareschutzes. Ich wünsche ihm die notwendige Beachtung und hoffe, dass er viele weitere Forschende und Praktiker animiert, sich ebenfalls Gedanken zu diesem Thema zu machen. Wir werden diese Diskussion auf jeden Fall genau verfolgen und der Studie an verschiedenen Veranstaltungen eine Plattform bieten.

Ich danke allen beteiligten Autorinnen und Autoren für den wertvollen Beitrag, der sicher auch jungen Leuten zeigt, dass es nicht zuletzt dank eines adäquaten Schutzes lohnenswert ist, sich etwas mehr Kenntnisse in der Kunst des Programmierens anzueignen, als ich das in meiner Jugend getan habe.

Catherine Chammartin
Direktorin des Eidgenössischen Instituts für Geistiges Eigentum

Bern, Oktober 2024

Avant-propos

« For x = 1 to 10; Print 'Hello World'; Next x; » : voilà à quoi ressemblait le premier « programme informatique » que j'ai écrit dans ma jeunesse. Il faisait apparaître dix fois « Hello World » à l'écran. Je l'avoue, ma première tentative n'était pas follement créative. Elle n'aurait en aucun cas pu prétendre une protection par le droit d'auteur ni par un brevet. De nos jours, la création de logiciels professionnels requiert très souvent la mobilisation de nombreuses ressources. Seule la perspective de récupérer au moins les fonds engagés est une motivation suffisante pour consentir des investissements de cette envergure. C'est la raison pour laquelle il est nécessaire que les logiciels soient protégés aussi efficacement que possible. Nombreux sont d'avis que la solution actuelle basée sur les droits de brevets ou d'auteur est insatisfaisante. En outre, la progression rapide de l'emploi de l'intelligence artificielle générative, notamment dans le domaine de la génération de code, remet encore davantage en question ses fondements.

Le moment est dès lors particulièrement propice pour lancer une réflexion sur de nouvelles formes de protection des logiciels en adéquation avec l'évolution technologique actuelle. Nous avons donc mandaté Florent Thouvenin et Peter Picht, tous deux professeurs à la faculté de droit de l'Université de Zurich, pour réaliser une étude sur le sujet.

Le défi d'une proposition de ce genre est que la nouvelle méthode de protection doit être plus attrayante pour toutes les parties impliquées par rapport aux options précédentes, puisque celles-ci sont inscrites dans des accords internationaux très difficilement modifiables. La présente étude prévoit donc de créer un droit sui generis pour les logiciels qui repose sur les systèmes de protection existants du droit des brevets et d'auteur, mais qui réponde également aux besoins de protéger efficacement les logiciels.

Je vois dans cette proposition une contribution importante à la discussion naissante sur l'avenir de la protection des logiciels. J'espère qu'on lui accordera l'attention nécessaire et qu'elle incitera de nombreux autres chercheurs et professionnels à réfléchir sur ce sujet. Nous suivrons en tous les cas cette discussion de près et offrirons une plateforme à l'étude lors de différentes manifestations.

Je remercie toutes les autrices et tous les auteurs pour leur précieuse contribution qui montre certainement aux jeunes qu'avec une protection adéquate il n'est pas vain d'acquérir des connaissances plus approfondies dans l'art de la programmation, allant au-delà de mes balbutiements de jeunesse.

Catherine Chammartin
Directrice de l'Institut Fédéral de la Propriété Intellectuelle

Berne, octobre 2024

Prefazione

«For x=1 to 10; Print ‘Hello World’; Next x;»: così appariva il primo programma per computer che ho scritto da giovane e che faceva comparire sullo schermo la scritta «Hello World» dieci volte. Lo ammetto: la mia prima opera non era molto creativa e non avrebbe potuto essere protetta né dal diritto d’autore né dal diritto dei brevetti. Per sviluppare un software professionale oggi sono spesso necessarie molte risorse. Si è incentivati a fare questi investimenti solo se si prevede la possibilità di recuperare perlomeno i fondi investiti inizialmente. È quindi essenziale che la protezione del software sia il più efficiente possibile. L’attuale soluzione basata sul diritto dei brevetti e sul diritto d’autore è insoddisfacente per molti, e il rapido avanzamento dell’uso dell’intelligenza artificiale generativa, in particolare nell’ambito della generazione di codici, ne sta mettendo ulteriormente in questione le fondamenta.

È quindi un buon momento per pensare a nuove forme di protezione del software, che siano più adatte agli sviluppi tecnologici attuali. Pertanto abbiamo commissionato un corrispondente studio a Florent Thouvenin e Peter Picht, professori presso la facoltà di giurisprudenza dell’Università di Zurigo.

La sfida di una proposta di questo genere è che il nuovo metodo di protezione deve essere più attraente per tutte le parti coinvolte rispetto alle opzioni precedenti, poiché queste ultime sono sancite in trattati internazionali molto difficili da modificare. Questo contributo propone di creare un diritto sui generis per il software che, pur basandosi sui sistemi di protezione esistenti del diritto dei brevetti e del diritto d’autore, sia adatto a soddisfare l’esigenza di una protezione efficace del software.

Ritengo che questa proposta rappresenti un importante contributo al fiorente dibattito sul futuro della protezione del software. Mi auguro quindi che riceva la necessaria attenzione e che incoraggi molti altri ricercatori e professionisti a riflettere su questo tema. Seguiremo sicuramente da vicino questa discussione e presenteremo lo studio in occasione di diversi eventi.

Ringrazio tutti gli autori coinvolti per il loro prezioso contributo, che indubbiamente dimostra anche ai giovani che vale la pena, non da ultimo grazie a un’adeguata protezione, di acquisire un po’ più di conoscenze nell’arte della programmazione di quanto non abbia fatto io in gioventù.

Catherine Chammartin
Direttrice dell’Istituto Federale della Proprietà Intellettuale

Berna, ottobre 2024

Table of Contents

List of Abbreviations

§/§§	section(s)
A/S	Aktieselskab (joint-stock company)
AB	Aktiebolag (limited liability company)
AC	Appeal Cases
AI	Artificial Intelligence
AIPPI	The International Association for the Protection of Intellectual Property
Akron L. Rev.	Akron Law Review
All ER	All England Law Reports
Am. Econ. Rev.	American Economic Review
Am. J. Comp. L.	The American Journal of Comparative Law
Am. U. Int'l L. Rev.	American University International Law Review
Anwaltsrevue	Das Praxismagazin des Schweizerischen Anwaltsverbandes (The Swiss Bar Association's magazine for legal practice)
API	Application Programming Interfaces
APLR	Asia Pacific Law Review
Art./Arts.	Article/Articles
Asian J. Int'l L.	Asian Journal of International Law
B. C. L. Rev.	Boston College Law Review
BBl	Bundesblatt (Swiss Federal Gazette)
Beijing L. Rev.	Beijing Law Review
Beijing Treaty	Beijing Treaty on Audiovisual Performances
BGE	Bundegerichtsentscheid, Amtliche Sammlung (Decision of the Swiss Supreme Court, Official Gazette)
BGer	Schweizerisches Bundesgericht (Swiss Supreme Court)
BGH	Bundesgerichtshof (German Federal Court of Justice)
BPatG	Bundespatentgericht (German Federal Patent Court)
CCPA	The United States Court of Customs and Patent Appeals
CDPA	Copyright, Designs and Patents Act 1988
CFI	European Court of First Instance
Ch	Chancery Division
ch.	chapter
CII	computer-implemented inventions
Cir.	Circuit
cit.	citation
Civ	Civil Division
CJEU	Court of Justice of the European Union
CL & SR	Computer Law and Security Review
CLC	Copyright Law of the PRC
Co Ltd	Company Limited
Colo. Law.	Colorado Lawyer
Comm. of the ACM	Communications of the Association for Computing Machinery
Comput. L. Rev. & Tech.	Computer Law Review and Technology Law Journal
Computer L.J.	Computer/Law Journal
CONTU	National Commission on New Technological Uses of Copyrighted Works
CoO	Swiss Code of Obligations of 30 March 1911 (SR 220)
CopA	Swiss Federal Act on Copyright and Related Rights of 9 October 1992 (SR 231.1)
Corp	Corporation
CPU	central processing unit
CR	Computer und Recht (Computer and Law)
CRi	Computer Law Review International
DDC	District Court for the District of Columbia

Dir.	Directive
Dir. 2009/24	Directive 2009/24/EC of the European Parliament and of the Council of 23 April 2009 on the legal protection of computer programs (Codified version), OJ L 111, 5 May 2009, 16–22
Diss.	Dissertation
DSRITB	Deutsche Stiftung für Recht und Informatik (German Foundation for Law and Informatics)
DVPW	Deutschen Vereinigung für Politische Wissenschaft (German Association for Political Science)
E.D. Va.	Eastern District of Virginia
e.g.	exempli gratia (for example)
EBA	Enlarged Board of Appeal (EPO)
EC	European Commission
ECDR	European Copyright and Design Reports
ed.	editor
edn.	edition
eds.	editors
EEC	European Economic Community
Emory L.J.	Emory Law Journal
EPA	see EPO
EPC	European Patent Convention of 5 October 1973, revised in Munich on 29 November 2000 (SR 0.232.142.2)
EPO	European Patent Office
EPÜ	see EPC
et al.	et alia (and others)
et seq.	and the following
et seqq.	and those following
EU	European Union
EWCA	Court of Appeal of England and Wales
EWHC	High Court of England and Wales
EWS	Europäisches Wirtschafts- und Steuerrecht
F. Supp. 3d	Federal Supplement, Third Series
F.2d	Federal Reporter, Second Series
F.3d	Federal Reporter, Third Series
F.4th	Federal Reporter, Fourth Series
FCA	Federal Court of Australia
FCAFC	Federal Court of Australia – Full Court
Fed. Cir.	Federal Circuit
fn.	footnote
FSR	Fleet Street Reports
GDPR	General Data Protection Regulation
GNU GPL	GNU General Public License
GPT-4	Generative Pre-trained Transformer 4
GRUR Int.	Gewerblicher Rechtsschutz und Urheberrecht, Internationaler Teil (Industrial Property Protection and Copyright, International Section)
GRUR Patent	Gewerblicher Rechtsschutz und Urheberrecht, Patentrecht in der Praxis (Industrial Property Protection and Copyright, Patent Law in Practice)
GRUR	Gewerblicher Rechtsschutz und Urheberrecht (Industrial Property Protection and Copyright)
GRUR-Prax.	Gewerblicher Rechtsschutz und Urheberrecht, in der Praxis (Industrial Property Protection and Copyright, in Practice)
Gyo-Ke	The first instance of administrative litigation in high court (Japan)
Gyo-Tsu	Administrative litigation cases heard by the Supreme Court (Japan)
Gyo-U	The first instance of administrative litigation in district court (Japan)
Habil.	Habilitation
Harv. L. Rev.	Harvard Law Review

HCC	Human Choice and Computers
i.a.	inter alia (among other things)
i.c.w.	in conjunction with
i.e.	id est (that is)
i.S.	in Sachen (in matters)
IAM	Intellectual Asset Management
ICLR	International Conference on Learning Representations
IDEA	The Law Review of the Franklin Pierce Center for Intellectual Property
IEEE	Institute of Electrical and Electronics Engineers
IFIP	International Federation for Information Processing
IIC	International Review of Intellectual Property and Competition Law
IIP Bulletin	Institute of Intellectual Property Bulletin
Inc.	Incorporated
Informatik Forsch. Entw.	Informatik in Forschung und Entwicklung (Computer Science in Research and Development)
Int. J. Law Inf. Technol.	International Journal of Law and Information Technology
Int. Prop. Q.	Intellectual Property Law Quarterly
Int. Rev. Law Comput. Technol.	International Review of Law, Computers and Technology
Intellect. Prop. Q.	Intellectual Property Quarterly
IP	Intellectual Property
IPHC	Intellectual Property High Court (Japan)
IPI	Swiss Federal Institute of Intellectual Property
IRPLC	Implementing Regulations of the Patent Law of the People's Republic of China (amended up to December 11, 2023)
Issues Sci. Technol.	Issues in Science and Technology
IT	Information Technology
J. Econ. Behav. Organ.	Journal of Economic Behavior and Organization
J. Econ. Issues	Journal of Economic Issues
J. Econ. Perspect.	Journal of Economic Perspectives
J. Knowl. Econ.	Journal of the Knowledge Economy
J. Leg. Stud.	Journal of Legal Studies
J. World Intellect. Prop.	Journal of World Intellectual Property
JEP	Journal of Economic Perspectives
JIPBA	Japanese Intellectual Property Basic Act
JIPITEC	Journal of Intellectual Property, Information Technology and Electronic Commerce Law
JIPLP	Journal of Intellectual Property Law & Practice
JPO	Japan Patent Office
Jurid. Trib.	Juridical Tribune
KI	see AI
LBA	Legal Board of Appeal (EPO)
LEM	Laboratory of Economics and Management
lit.	litera
LLP	Limited Liability Partnership
LSR	Life Science Recht
Ltd	Limited
M&A	Mergers and Acquisitions
Manag. Sci.	Management Science
Masaryk Univ. J. Law Technol.	Masaryk University Journal of Law and Technology
Mich. St. L. Rev.	Michigan State Law Review
Mich. Telecomm. & Tech. L. Rev.	Michigan Telecommunications and Technology Law Review
MIT	Massachusetts Institute of Technology
MMR	Multimedia und Recht, Zeitschrift für IT-Recht und Recht der Digitalisierung (Multimedia and Law, Journal for IT Law and Digitalization Law)
MSME	micro, small or medium-size enterprise

Ne	Appeal from district court in civil litigation (Japan)
NJW	Neue Juristische Wochenschrift (New Weekly Legal Journal)
NLP	Natural Language Processing
OECD	Organization for Economic Co-operation and Development
OFK	Orell Füssli Kommentar (Orell Füssli Commentary)
OGer	Obergericht (High Court)
OJ	Official Journal
para./paras.	paragraph/paragraphs
PAS	Patents Act 1994 of the Republic of Singapore (No. 21 of 1994)
Pat. & Licensing	Patents & Licensing
Pat/pat	Patents Court/patent
PatA	Swiss Federal Act on Patents for Inventions of 25 June 1954 (SR 232.14)
PatG	Patentgesetz (Patent Act)
PIJIP	Program on Information Justice and Intellectual Property
PLC	Patent Law of the People's Republic of China
PRC	People's Republic of China
Pre.	Preliminary remarks
Pte Ltd	Private Limited
R&D	Research and Development
RAND J. of Econ.	The RAND Journal of Economics
RCSP	Regulations on Computer Software Protection of the PRC of 1 January 2002
RD	Re-examination decision
RDi	Recht Digital (Law Digital)
Res. Policy	Research Policy
RICL	Regulations for the Implementation of the Copyright Law of the PRC
RPC	Reports Patent Cases
RSDA	Revue suisse de droit des affaires et du marché financier (Swiss Review of Business and Financial Market Law)
S. Cal. L. Rev.	Southern California Law Review
S. Ct.	Supreme Court
SAcLJ	Singapore Academy of Law Journal
Sci. Comput. Program.	Science of Computer Programming
SCJ	Supreme Court of Japan
SCRIPTed	Journal of Law, Technology & Society
Sdn Bhd	Sendirian Berhad (Private Limited)
Sec.	section
SGCA	Singapore Court of Appeal
SGHC	Singapore High Court
SGHCR	Singapore High Court Registrar
SHK	Stämpflis Handkommentar (Stämpfli's Commentary)
sic!	Zeitschrift für Immaterialgüter-, Informations- und Wettbewerbsrecht (Journal of Intellectual Property, Information and Competition Law)
SIM	subscriber identity module
SIWR	Schweizerisches Immaterialgüter- und Wettbewerbsrecht (Swiss Intellectual Property and Competition Law)
SJZ	Schweizerische Juristen-Zeitung (Swiss Lawyers' Journal)
SLR	Singapore Law Reports
SLR(R)	Singapore Law Reports Reissue
SOFTIC	Software Information Center
SR	Systematische Rechtssammlung (Systematic Legal Collection)
St. John's L. Rev	St. John's Law Review
Stanford L. Rev.	Stanford Law Review
SWE	Software Engineering Bench
TBA	Technical Board of Appeal (EPO)

TDM	Text and Data Mining
Texas L. Rev.	Texas Law Review
ToA	Federal Act on the Protection of Topographies of Semiconductor Products of 9 October 1992 (SR 231.2)
ToG	see ToA
TRIPS	Agreement on Trade-Related Aspects of Intellectual Property Rights
U. Chi. L. Rev.	The University of Chicago Law Review
U. Pa. L. Rev.	University of Pennsylvania Law Review
U.S.C.	United States Code
UCLA Pac. Basin L.J.	University of California Pacific Basin Law Journal
UK	United Kingdom
UKHL	United Kingdom House of Lords
UKIPO	Intellectual Property Office UK
UKSC	United Kingdom Supreme Court
URG	see CopA
UrhG	German Copyright Act of 9 September 1965 (BGBl. I S. 1273)
US/U.S.	United States
USCO	U.S. Copyright Office
USPTO	United States Patent and Trademark Office
v.	versus
Va. J.L. & Tech.	Virginia Journal of Law and Technology
Va. L. Rev.	Virginia Law Review
viz.	videre licet
VLSI	Very-Large-Scale Integrated Circuit
w.f.r.	with further references
Wa	The first instance of civil litigation in district court (Japan)
WCT	WIPO Copyright Treaty
WIPO	World Intellectual Property Organization
wipr	World International Property Review
WLR	Weekly Law Reports
WPPT	WIPO Performances and Phonograms Treaty
WTO	World Trade Organization
ZH	Zurich
ZUM	Zeitschrift für Urheber- und Medienrecht (Journal of Copyright and Media Law)

List of References

ABBOTT RYAN, I Think, Therefore I Invent: Creative Computers and the Future of Patent Law, B.C. L. Rev. 2016, Vol. 57, No. 4, 1079–1126

AMANKWAH-AMOAH JOSEPH/KEHINDE MEDASE STEPHEN, Extracting Innovation Value from Intellectual Property: Evidence from sub-Saharan Africa, J. Knowl. Econ. 2023, 1–35

ANDERMATT ADRIAN, Die arbeitsrechtliche Zuordnung von immaterialgüterrechtlich geschützten Arbeitsergebnissen, SJZ 2008, Vol. 104, No. 12, 285–293

ANN CHRISTOPH, Patentrecht, Lehrbuch zum deutschen und europäischen Patentrecht und Gebrauchsmusterrecht, 8th edn., Munich 2022

AOKI HIROYA, Discussion of Copyright Protection for AI-created Works in Japan, in: Mendoza-Caminade Alexandra (ed.), L'Entreprise et l'Intelligence Artificielle – les Réponses du Droit, Toulouse 2022, 269–276

ASAY CLARK, Software's Copyright Anticommons, Emory L.J. 2017, Vol. 66, No. 2, 265–331

ASUNCION ESTEVE, Patent Protection of Computer-Implemented Inventions Vis-À-Vis Open Source Software, J. World Intellect. Prop. 2006, Vol. 7, No. 3, 276–300

ATILLA SÖĞÜT, Dealing with AI-generated works: lessons from the CDPA section 9(3), JIPLP 2024, Vol. 19, No. 1, 43–54

BACCELLI MICHELE/KOBAYASHI KAZUTO/SEREBOFF STEVEN C./HIRATSUKA MITSUYOSHI, Advanced Software and Patents: A Patentability Balance for Fostering Technology, 14th IFIP International Conference on Human Choice and Computers (HCC), Tokyo 2020 (cit.: BACCELLI et al.)

BALDUS OLIVER, Decision G 1/19 and the Messy Misconception of the COMVIK Approach, GRUR Int. 2021, Vol. 70, No. 10, 957–962

BALLARDINI ROSA MARIA, The Software Patent Thicket: A Matter Of Disclosure, SCRIPTed 2009, Vol. 6, No. 2, 207–233

BARENKAMP MARCO/REBSTADT JONAS/THOMAS OLIVER, Applications of AI in classical software engineering, AI Perspectives & Advances 2020, Vol. 1, No. 2, 1–15 (cit.: BARENKAMP et al., AI Perspectives & Advances 2020)

BARRELET DENIS/EGLOFF WILLI (eds.), Das neue Urheberrecht, Kommentar zum Bundesgesetz über das Urheberrecht und verwandte Schutzrechte, 4th edn., Bern 2020 (cit.: AUTHOR, in: Barrelet/Egloff)

BATARSEH FERAS A./MOHOD RASIKA/KUMAR ABHINAV/BUI JUSTIN, The Application of artificial intelligence in software engineering: a review challenging conventional wisdom, in: Batarseh Feras A./Yang Ruixin, Data Democracy (eds.), At the Nexus of Artificial Intelligence, Software Development, and Knowledge Engineering, London 2020, 179–232 (cit.: BATARSEH et al.)

BAUMANN MALTE, Generative KI und Urheberrecht – Urheber und Anwender im Spannungsfeld, Vol. 76, No. 51, NJW 2023, 3673–3744

BENKARD INGO/EHLERS JOCHEN (eds.), EPÜ Kommentar Benkard, 4th edn., Munich 2023 (cit.: AUTHOR, in: EPÜ Kommentar Benkard)

BENNETT RICHARD, The Future of Protecting Computer-Implemented Methods of Simulation under the European Patent Convention, sic! 2020, No. 5, 240–245

BENTLY LIONEL, SHERMAN BRAD, GANGJEE DEV, JOHNSON PHILLIP, Intellectual Property Law, 6th edn., Oxford 2022 (cit.: BENTLY et al.)

BERGER MATHIAS, Schutz von Software – Überblick über die Rechtslage in der Schweiz, in: Trüeb Hans Rudolf (ed.), Softwareverträge: Referate der Tagung der Stiftung für juristische Weiterbildung Zürich vom 11. November 2003, Zurich 2004, 25–60

BERGSTRA JAN A./KLINT PAUL, About "trivial" software patents: The IsNot case, Sci. Comput. Program. 2007, Vol. 64, No. 3, 264–285

BESEN STANLEY M./RASKIND LEO J., An Introduction to the Law and Economics of Intellectual Property, J. Econ. Perspect. 1991, Vol. 5, No. 1, 3–27

BESSEN JAMES/HUNT ROBERT M., The Software Patent Experiment, in: Patents, Innovation and Economic Performance, OECD Conference Proceedings, Paris 2004

BESSEN JAMES/MASKIN ERIC, Sequential innovation, patents, and imitation, RAND J. of Econ. 2004, Vol. 40, No. 4, 611–635

BOLGER MUIREANN, UK govt confirms backtrack on 'broad exceptions' for AI, wipr, 1 January 2023, available at: https://www.worldipreview.com/copyright/uk-govt-confirms-backtrack-on-broad-exceptions-for-ai-24623 (last accessed: 19 July 2024).

BONADIO ENRICO/MCDONAGH LUKE/DINEV PLAMEN, Artificial intelligence as inventor: exploring the consequences for patent law, Intellect. Prop. Q. 2021, Vol. 1, 48–66 (cit.: BONADIO et al., Intellect. Prop. Q. 2021)

BOND TOBY/BLAIR SARAH, Artificial Intelligence & Copyright: Section 9(3) or authorship without an author, JIPLP 2019, Vol. 14, No. 6, 423

BÖTTCHER HORST, Die urheberrechtliche Erschöpfung und ihre Bedeutung im digitalen Umfeld, Diss., Bern 2013

BUBECK SÉBASTIEN/CHANDRASEKARAN VARUN/ELDAN RONEN/GEHRKE JOHANNES/HORVITZ ERIC HORVITZ/KAMAR ECE/LEE PETER/LEE YIN TAT/LI YUANZHI/LUNDBERG SCOTT/NORI HARSHA/PALANGI HAMID/TULIO RIBEIRO MARCO/ZHANG YI, Sparks of Artificial General Intelligence: Early experiments with GPT-4, 13 April 2023, available at: https://arxiv.org/abs/2303.12712 (last accessed: 19 July 2024), (cit.: BUBECK et al.)

BUDISH ERIC, ROIN BENJAMIN N., WILLIAMS HEIDI, Patents and Research Investments: Assessing the Empirical Evidence, Am. Econ. Rev. 2016, Vol. 106, No. 5, 183–187 (cit.: BUDISH et al., Am. Econ. Rev. 2016)

BUSCHE JAN/STOLL PETER-TOBIAS/WIEBE ANDREAS (eds.), TRIPs Internationales und europäisches Recht des geistigen Eigentums, 2nd edn., Cologne 2013 (cit.: AUTHOR, in: Busche et al.)

BUSSE RUDOLF/KEUKENSCHRIJVER ALFRED (eds.), PatG Patentgesetz, 9th edn., Berlin/Boston 2020 (cit.: AUTHOR, in: Busse/Keukenschrijver)

CALAME THIERRY, Die Wirkung des Patents, in: Büren Roland/David Lucas M. (eds.), Patentrecht und Know-how, unter Einschluss von Gentechnik, Software und Sortenschutz, SIWR IV, Basel 2006

CAMPBELL-KELLY MARTIN, Not All Bad: An Historical Perspective on Software Patents, Mich. Telecomm. & Tech. L. Rev. 2005, Vol. 11, No. 2, 191–248

CHEN HAO-YUN, Copyright Protection for Software 2.0?, Rethinking the Justification of Software Protection under Copyright Law, in: Lee Jyh-An/Hilty Reto/Liu Kung-Chung (eds.), Artificial Intelligence and Intellectual Property, Oxford 2021, 323–340

CHERPILLOD IVAN, Intelligence artificielle et droit d'auteur, sic! 2023, No. 9, 445–452

CHERPILLOD IVAN, Schranken des Urheberrechts, in: SIWR II/1, Urheberrecht und verwandte Schutzrechte, 3rd ed., Basel 2014 (cit.: CHERPILLOD, SIWR II/1)

CLABURN THOMAS, GitHub, Microsoft, OpenAI fail to wriggle out of Copilot copyright lawsuit – Judge won't toss out two key charges, software source slurping case still on Initial Complaint available, The Register, 12. May 2023, available at: https://www.theregister.com/2023/05/12/github_microsoft_openai_copilot/ (last accessed: 19 July 2024)

DAI ZHE/JIN BANGGUI, The copyright protection of AI-generated works under Chinese law, Jurid. Trib. 2023, Vol. 13, No. 2, 241–260

DAPP MARCUS M., The effects of software patent policy on the motivation and innovation of free and open source software developers, Diss., Zurich 2009

DAVIES GILLIAN, Copyright and the Public Interest, 2nd edn., London 2002

DAVIS RICHARD/ST. QUINTIN THOMAS/TRITTON GUY (eds.), Tritton on Intellectual Property in Europe, 6th edn., London 2022 (cit.: DAVIS et al.)

DE LA DURANTAYE, "Garbage in, garbage out" – Die Regulierung generativer KI durch Urheberrecht, ZUM 2023, Vol. 67, No. 10, 645–660

DE RASSENFOSSE GAÉTAN/JAFFE ADAM/WASSERMAN MELISSA, AI-Generated Inventions: Implications for the Patent System, S. Cal. L. Rev. 2023, Vol. 96, No. 6, 101–126 (cit.: DE RASSENFOSSE et al., S. Cal. L. Rev. 2023)

DE RASSENFOSSE GAÉTAN/PALANGKARAYA ALFONS/WEBSTER ELIZABETH, Why Do Patents Facilitate Trade in Technology? Testing the Disclosure and Appropriation Effects, Res. Policy 2016, Vol. 43, No. 7, 1326–1336 (cit.: DE RASSENFOSSE et al., Res. Policy 2016)

DE WERRA JACQUES/GILLIERON PHILIPPE (eds.), Commentaire romand, Propriété intellectuelle, Basel 2013 (cit.: AUTHOR, in: Commentaire romand)

DERMAWAN ARTHA, Text and data mining exceptions in the development of generative AI models: What the EU member states could learn from the Japanese "nonenjoyment" purposes?, J. World Intellect. Prop. 2023, Vol. 27, No. 1, 1–25

DORNIS TIM W., Künstliche Intelligenz und Design, in: Dönch Julia/Zentek Sabine(eds.), DesignG: Designgesetz mit Gemeinschaftsgeschmacksmusterrecht, Baden-Baden 2022 (cit.: DORNIS, KI und Design)

DORNIS TIM W., Künstliche Intelligenz und Patentrecht – "Klarstellungen zur Erfindung ohne Erfinder", GRUR Patent 2023, Vol. 1, No. 1, 14–19

DOSI GIOVANNI/PALAGI ELISA/ROVENTINI ANDREA/RUSSO EMANUELE, Do patents really foster innovation in the pharmaceutical sector? Results from an evolutionary, agent-based model, J. Econ. Behav. Organ. 2023, Vol. 212, 564–589 (cit.: DOSI et al., J. Econ. Behav. Organ.)

DUFFUS ANNA, The Proposal for a Directive on the Patentability of Computer-implemented Inventions, Int. Rev. Law Comput. Technol. 2002, Vol. 16, No. 3, 331–338

DURNEY EDWARD G., Protection of Computer Programs under Japanese Copyright t Law, UCLA Pac. Basin L.J. 1991, Vol. 9, No. 1-2, 17–77.

EAGLES IAN/LONGDIN LOUISE, Technological Creativity and Moral Rights: A Comparative Perspective, Int. J. Law Inf. Technol. 2023, Vol. 12, No. 2, 209–236

EBERS MARTIN/HEINZE CHRISTIAN/KRÜGEL TINA/STEINRÖTTER BJÖRN (eds.), Künstliche Intelligenz und Robotik Rechtshandbuch, Munich 2020 (cit.: AUTHOR, in: KI-Rechtshandbuch)

ECKL JULIAN, Das Scheitern des Vorschlags für eine EU-Richtlinie zur 'Patentierbarkeit computerimplementierter Erfindungen': Einige Erklärungen und eine Bewertung, Papier zur Tagung "Governing the Knowledge Society" des Arbeitskreises Politische Steuerung der DVPW in Hamburg, 2006

ENGEL ANDREAS, Can a Patent Be Granted for an AI-Generated Invention?, GRUR Int. 2020, Vol. 69, No. 11. 1123–1129

ERTEL WOLFGANG, Grundkurs Künstliche Intelligenz, Eine praxisorientierte Einführung, 5th edn., Wiesbaden 2021

EVANS DAVID S./LAYNE-FARRAR ANNE, Software Patents and Open Source: The Battle Over Intellectual Property Rights, Va. J.L. & Tech. 2004, Vol. 9, No. 3, 1–28

FOWLER MARTIN, Refactoring, Improving the Design of Existing Code, 2nd edn., Boston 2019

FROMER JEANNE C., Expressive Incentives in Intellectual Property, Va. L. Rev. 2012, Vol. 98, No. 8, 1746–1824

FUCHS THOMAS, Ansätze für einen Interessenausgleich im Softwarepatentrecht, 2004, available at: https://delegibus.com/2004,5.pdf (last accessed: 19 July 2024)

FURRER ANDREAS, Die Einbettung von Smart Contracts in das schweizerische Privatrecht, Anwaltsrevue 2018, Vol. 21, No. 3, 103–115

GAFFAR HAFIZ/ALBARASHDI, Copyright Protection for AI-Generated Works: Exploring Originality and Ownership in a Digital Landscape, Asian J. Int'l L. 2024, 1–24

GAJECK NICLAS A./SCHEIBE ROMAN C., KI-Schutz im Erfinderrecht – die Lage nach DABUS, RDi 2023, Vol. 3, No. 9, 408–414

GANS JOSHUA S./HSU DAVID H./STERN SCOTT, The Impact of Uncertain Intellectual Property Rights on the Market for Ideas: Evidence from Patent Grant Delays, Manag. Sci. 2008, Vol. 54, No. 5, 982–997 (cit.: GANS et al., Manag. Sci. 2008)

GARFINKEL SIMON L./STALLMAN RICHARD M./KAPOR MITCHELL, Why Patents Are Bad for Software, Issues Sci. Technol. 1991, Vol. 8, No. 1, 50–55 (cit.: GARFINKEL et al., Issues Sci. Technol. 1991)

GEIGER CHRISTOPHE/GERVAIS DANIEL/SENFTLEBEN MARTIN, The Three-Step Test Revisited: How to Use the Test's Flexibility in National Copyright Law, PIJIP 2013, No. 4, 1–44 (cit.: GEIGER et al., PIJIP 2013)

GEIGER CHRISTOPHE/GRIFFITHS JONATHAN/HILTY RETO M., Erklärung für eine ausgewogene Auslegung des Drei-Stufen-Tests im Urheberrecht, GRUR Int. 2008, Vol. 57, No. 10, 822–825 (cit.: GEIGER et al., GRUR Int. 2008)

GINSBURG JANE C., Copyright, in: Dreyfuss Rochelle/Pila Justine (eds.), Oxford Handbook of Intellectual Property Law, Oxford 2017, 487–516

GRATTON ÉLOÏSE, Should Patent Protection Be Considered for Computer Software-Related Innovations?, Comput. L. Rev. & Tech. J. 2003, Vol. 7, No. 2, 223–253

GROB MATTHIAS, Patentierbarkeit computerimplementierter Simulationen und die Entscheidung G 1/19, GRUR-Prax 2021, Vol. 13, No. 9, 243–245

GROSCHE ANDREAS, Software Patents - Boon or Bane for Europe? Int. J. Law Inf. Technol. 2006, Vol. 14, No. 3, 257–309

GRUNER RICHARD S., Better Living Through Software: Promoting Information Processing Advances Through Patent Incentives, St. John's L. Rev. 2000, Vol. 74, No. 4, 977–1068

GRYNBAUM MICHAEL M./MAC RYAN, The Times Sues OpenAI and Microsoft Over A.I. Use of Copyrighted Work: Millions of articles from The New York Times were used to train chatbots that now compete with it, the lawsuit said, The New York Times, 27 December 2023, available at: https://www.nytimes.com/2023/12/27/business/media/new-york-times-open-ai-microsoft-lawsuit.html (last accessed: 19 July 2024)

GUADAMUZ ANDRES, Do Androids Dream of Electric Copyright? Comparative Analysis of Originality in Artificial Intelligence Generated Works, in: Lee Jyh-An/Hilty Reto/Liu Kung-Chung (eds.), Artificial Intelligence and Intellectual Property, Oxford 2021, 147–176 (cit.: GUADAMUZ, Originality)

GUADAMUZ ANDRES, The Software Patent Debate, JIPLP 2006, Vol. 1, No. 3, 96–206

HA TRINA/TEH JOO LIN/ALLEN JASON GRANT/KARUPPIAH BYRON/PENG HUIJUA, When Code Creates: A Landscape Report on Issues at the Intersection of Artificial Intelligence and Intellectual Property Law, IPOS (2024), available at: https://www.ipos.gov.sg/resources/publications (last accessed: 19 July 2024), (cit.: HA et al.)

HARMAN MARK, The Role of Artificial Intelligence in Software Engineering, IEEE, 2012 First International Workshop on Realizing AI Synergies in Software Engineering (RAISE), Zurich 2012, 1–6

HART ROBERT/HOLMES PETER/REID JOHN, The Economic Impact of Patentability of Computer Programs, Report to the European Commission, Study Contract ETD/99/B5-3000/E/106 (2000), (cit.: HART et al.)

HARTMANN DAMIAN, Text and Data Mining and Copyright in Switzerland and the European Union, sic! 2023, No. 3, 157–167

HE TIANXIANG, Copyright Exceptions Reform and AI Data Analysis in China, A Modest Proposal, in: Lee Jyh-An/Hilty Reto/Liu Kung-Chung (eds.), Artificial Intelligence and Intellectual Property, Oxford 2021, 196–220 (cit.: HE, Copyright Exceptions Reform)

HE TIANXIANG, The sentimental fools and the fictitious authors: rethinking the copyright issues of AI-generated contents in China, APLR 2020, Vol. 27, No. 2, 218-238

HECKEL PAUL, Debunking the Software Patent Myths, Comm. of the ACM 1992, Vol. 35, No. 6, 121–140

HEGDE DEEPAK/LUO HONG, Patent Publication and the Market for Ideas, Manag. Sci. 2018, Vol. 64, No. 2, 652–672

HEINRICH PETER, PatG/EPÜ, Schweizerisches Patentgesetz/Europäisches Patentübereinkommen, Kommentar in synoptischer Darstellung, 3rd edn., Bern 2018 (cit.: HEINRICH, in: PatG/EPÜ Kommentar)

HILTI CHRISTIAN/KÖPF ALFRED/STAUBER DEMIAN/CARREIRA ANDREA (eds.), Schweizerisches und europäisches Patent- und Patentprozessrecht, 4th edn., Bern 2021 (cit.: AUTHOR, in: Hilti et al.)

HILTY RETO M., "Bildungssoftware" – Obergericht Zürich vom 24. Januar 2013, sic! 2013, No. 11, 697–707

HILTY RETO M., Die Rechtsnatur des Softwarevertrages, CR 2012, Vol. 28, No. 10, 625–637

HILTY RETO M., Urheberrecht, 2nd edn., Bern 2020 (cit.: HILTY, Urheberrecht)

HILTY RETO M./GEIGER CHRISTOPHE, Patenting Software? A Judicial and Socio-Economic Analysis, IIC 2005, Vol. 36, No. 6, 615–646

HILTY RETO M./HOFFMANN JÖRG/SCHEUERER STEFAN, Intellectual Property Justification for Artificial Intelligence, in: Lee Jyh-An/Hilty Reto/Liu Kung-Chung (eds.), Artificial Intelligence and Intellectual Property, Oxford 2021, 50–72 (cit.: HILTY et al.)

HOEREN THOMAS/SPITTKA JAN, Patentschutz für Software – zum Streit über die Richtlinie über die Patentierbarkeit computerimplementierter Erfindungen, Ad Legendum 2005, Vol. 2, No. 3, 171–175

HOEREN THOMAS/WEHKAMP NILS, Individualität im Quellcode? Softwareschutz und Urheberrecht, CR 2018, Vol. 34, No. 1, 1–7

HRISTOV KALIN, Artificial Intelligence and the Copyright Dilemma, IDEA 2017, Vol. 57, No. 3, 437–454

HUBMANN HEINRICH, Das Recht des schöpferischen Geistes. Eine philosophisch-juristische Betrachtung zur Urheberrechtsreform, Berlin 1954

IGBOKWE EZINNE MIRIAN, Human to machine innovation: Does legal personhood and inventorship threshold offer any leeway?, J. World Intellect. Prop. 2024, 1–26

ISHIDA MASAYASU, Outline of the Japanese Copyright Law, JPO 2008, available at: https://www.jpo.go.jp/e/news/kokusai/developing/training/textbook/ (last accessed: 19 July 2024).

ISLER MICHAEL, Text und Data Mining in der medizinischen Forschung, LSR 2022, Vol. 5, No. 2, 111–119

JAFFE ADAM B., The U.S. patent system in transition: policy innovation and the innovation process, Res. Policy 2000, Vol. 29, No. 4/5, 531–557

JIMENEZ CARLOS E./YANG JOHN/WETTIG ALEXANDER/YAO SHUNYU/PEI KEXIN/PRESS OFIR/NARASIMHAN KARTHIK, SWE-agent: Agent-Computer Interfaces Enable Automated Software Engineering, ICLR Conference Paper 2024 (cit.: JIMENEZ et al., ICLR Conference Paper 2024)

KARJALA DENNIS S./SUGIYAMA KEIJI, Fundamental Concepts in Japanese and American Copyright Law, Am. J. Comp. L. 1987, Vol. 36, No. 4, 613–679

KIM DARIA/DREXL JOSEF/HILTY RETO M./SLOWINSKI PETER R., Artificial Intelligence Systems as Inventors? A Position Statement of 7 September 2021 in view of the evolving case-law worldwide, Max Planck Institute for Innovation and Competition Research Paper 2021 (cit.: KIM et al.)

KMENT MARTIN/BORCHERT SOPHIE, Künstliche Intelligenz und Algorithmen in der Rechtsanwendung, Munich 2022

KONERTZ ROMAN/SCHÖNHOF RAOUL, Erfindungen durch Computer und künstliche Intelligenz – eine aktuelle Herausforderung für das Patentrecht, ZGE 2018, Vol. 10, No. 4, 379–411

KONISHI KAY, TAKAISHI HIDEKI, YOSHIDA ETSUKO, TANIGUCHI NOBUYUKI, FURUHASHI NOBUSHIGE, HORIKAWA TAMAKI, TSUNO MASAYA, AIPPI Group Report, Inventorship of Inventions made using Artificial Intelligence, 2020; available at: https://aippi.soutron.net/Portal/Default/en-GB/Search/SimpleSearch (last accessed: 19 July 2024), (cit.: KONISHI et al.)

KOSTOLANSKY KRIS J./SALGADO DANIEL A., Does the Experimental Use Exception in Patent Law Have a Future?, Colo. Law. 2018, No. 1, 32–40

KRAUSEN JEAN-MARCEL, Künstliche Intelligenz als die letzte Erfindung des Menschen?, GRUR 2023, Vol. 125, No. 12, 841–848

KREUTZER RALF T./SIRRENBERG MARIE, Künstliche Intelligenz verstehen, Grundlagen – Use-Cases – unternehmenseigene KI-Journey, Wiesbaden 2019

KRÜGER STEFAN/BIEHLER MANUEL/APEL SIMON, Keine "Used Games" aus dem Netz Unanwendbarkeit der "UsedSoft"-Entscheidung des EuGH auf Videospiele, MMR 2013, Vol. 16, No. 12, 760–765

KUDO TOSHITAKA, Criteria for the Recognition of Inventors and the Procedure to Settle Disputes about the Recognition of Inventors, IIP Bulletin 2006, No. 24, 202–210, available at: https://www.iip.or.jp/e/fellow/researcher_fy17_2005.html (last accessed: 19 July 2024)

LANDES WILLIAM/POSNER RICHARD, An Economic Analysis of Copyright Law, J. Leg. Stud. 1989, Vol. 18, No. 2, 325–363

LATINOVIĆ MILAN/PAMMER-SCHINDLER VIKTORIA, Automation and Artificial Intelligence: in Software Engineering: Experiences, Challenges, and Opportunities, Proceedings of the 54th Annual Hawaii International Conference on System Sciences (2021), available at: http://hdl.handle.net/10125/70628 (last accessed: 19 July 2024)

LAUB CHRISTOPH, Patentfähigkeit von Softwareerfindungen: Rechtliche Standards in Europa und in den USA und deren Bedeutung für den internationalen Anmelder, GRUR Int. 2006, Vol. 55, No. 8/9, 629–640

LEE JYH-AN, Copyright Divisibility and The Anticommons, Am. U. Int'l L. Rev. 2016, Vol. 32, No. 1, 117–163

LEINS-ZURMUEHLE SARAH, Ideation, Conceptualization, Realization, Discovering the Creative Scope in Software Engineering from the Perspective of Copyright and Patent Law, Diss., Zurich 2021

LEISNER WALTER GEORG, Das Scheitern der "Software-Patent-Richtlinie" – Was nun? EWS 2005, No. 9, 396–400

LEJEUNE MATHIAS/SIECKMANN RALF, Softwarepatente in den USA und die aktuelle Entwicklung in Deutschland und der EU – Analyse der Entscheidung des U.S. Supreme Court i.S. Bilski v. Kappos, MMR 2010, Vol. 13, No. 11, 741–745

LEMLEY MARK A./SHAPIRO CARL, Patent Holdup and Royalty Stacking, Texas L. Rev. 1991, Vol. 85, No. 7, 1991–2049

LERNER JOSHUA, The Importance of Patent Scope: An Empirical Analysis, RAND J. of Econ. 1994, Vol. 25, No. 2, 319–333

LEUPOLD ANDREAS/WIEBE ANDREAS/GLOSSNER SILKE (eds.), IT-Recht, Recht, Wirtschaft und Technik der digitalen Transformation, 4th edn., Munich 2021 (cit.: AUTHOR, in: Leupold et al.)

LI YUJIA/CHOI DAVID/CHUNG/KUSHMAN NATE/SCHRITTWIESER JULIAN/LEBLOND RÉMI/ECCLES TOM /KEELING JAMES/GIMENO FELIX/DAL LAGO AGUSTIN/HUBERT THOMAS/CHOY PETER/DE MASSON D'AUTUME CYPRIEN/BABUSCHKIN IGOR/CHEN XINYUN/HUANG PO-SEN/WELBL JOHANNES/GOWAL SVEN/CHEREPANOV ALEXEY/MOLLOY JAMES/MANKOWITZ DANIEL J./SUTHERLAND ROBSON ESME/ KOHLI PUSHMEET/DE FREITAS NANDO/KAVUKCUOGLU KORAY/VINYALS ORIOL, Competition-level code generation with AlphaCode, Science 2022, Vol. 379, No. 6624, 1092–1097 (cit.: LI et al.)

LIM DARYL, AI & IP Innovation & Creativity in an Age of Accelerated Change, Akron L. Rev. 2019, Vol. 52 No. 3, 813–875

LU TIAN, DABUS in China: So far, it is 0–2, IPKat 2024, 17 May 2024, available at: https://ipkitten.blogspot.com/2024/05/dabus-in-china-so-far-it-is-02.html (last accessed: 19 July 2024)

LUGINBÜHL STEFAN/GANEA PETER, Patent Law in Greater China, Cheltenham, England/Northampton, Massachusetts 2014 (cit.: AUTHOR, in: Luginbühl/Ganea)

LUTZ MARTIN J., Der Schutz der Computerprogramme in der Schweiz, GRUR Int. 1993, Vol. 42, No. 8/9, 653–663

MALEVANNY NIKITA, Die UsedSoft-Kontroverse: Auslegung und Auswirkungen des EuGH-Urteils, CR 2013, Vol. 29, No. 7, 422–427

MARBACH EUGEN/DUCREY PATRIK/WILD GREGOR, Immaterialgüter- und Wettbewerbsrecht, 4th edn., Bern 2017 (cit.: MARBACH et al.)

MARMY-BRÄNDLI SANDRA/OEHRI ISABELLE, Das Training künstlicher Intelligenz, sic! 2023, No. 12, 655–666

MARTINI MARIO, Blackbox Algorithmus – Grundfragen einer Regulierung Künstlicher Intelligenz, Berlin 2019

MAY CHRISTOPHER, A Global Political Economy of Intellectual Property Rights, The new enclosures?, 2nd edn., Abingdon, Oxfordshire 2010

MAZZOLENI ROBERTO/NELSON RICHARD, Economic Theories about the Benefits and Costs of Patents, J. Econ. Issues 1998, Vol. 32, No. 4, 1031–1052

MAZZOLENI ROBERTO/NELSON RICHARD, The benefits and costs of strong patent protection: a contribution to the current debate, Res. Policy 1998, Vol. 27, No. 3, 273–284

MEITINGER THOMAS, Smart Contracts, Informatik Spektrum 2017, Vol. 40, No. 4, 371–375

MENELL PETER S., Tailoring Legal Protection for Computer Software, Stanford L. Rev. 1987, Vol. 39, No. 6, 1329–1372

MÉNIÈRE YANN/PIHLAJAMAA HELI, Künstliche Intelligenz in der Praxis des EPA, GRUR 2019, Vol. 121, No. 4, 332–336

MILLER ARTHUR R., Copyright Protection for Computer Programs, Databases, and Computer-Generated Works: Is Anything New Since CONTU?, Harv. L. Rev. 1993, Mar., Vol. 106, No. 5, 977–1073

MINSSEN TIMO/ABOY MATEO, The Patentability of Computer-Implemented Simulations and Implications for Computer-Implemented Inventions (CIIs), JIPLP 2021, Vol. 16, No. 7, 633–635

MITRA-KAHN BENJAMIN, Economic reasons to recognise AI Inventors, in: Ryan Abbott (ed.), Research Handbook on Intellectual Property and Artificial Intelligence, Cheltenham 2022, 376–390

MÜLLER BARBARA K./OERTLI REINHARD (eds.), Urheberrechtsgesetz, Stämpflis Handkommentar, 2nd edn., Bern 2012 (cit.: AUTHOR, in: SHK)

MUÑOZ FERRANDIS CARLOS/DUQUE LIZARRALDE MARTA, Open sourcing AI: intellectual property at the service of platform leadership, JIPITEC 2022, Vol. 13, No. 3, 224–246

NACK RALPH, AIPPI Group Report, Inventorship of inventions made using Artificial Intelligence (2020), Q272-GR-P-2020 available at: https://aippi.soutron.net/Portal/Default/en-GB/Search/SimpleSearch (last accessed: 19 July 2024)

NÄGERL JOEL/NEUBURGER BENEDIKT/STEINBACH FRANK, Künstliche Intelligenz: Paradigmenwechsel im Patentsystem, GRUR 2019, Vol. 121, No. 4, 336–341 (cit.: NÄGERL et al.)

NAKAYAMA NOBUHIRO, Patent Act, Law Lectures Series, 3rd edn., Koubundou 2016

NEFF EMIL F./ARN MATTHIAS/LÜCK GERT, Urheberrecht im EDV-Bereich, in: SIWR II/2, Basel 1998 (cit.: NEFF et al.)

NG CHONG YUAN, Whiter Singapore's Harbour for Patenting Computer-Implemented Inventions?, SAcLJ 2021, Vol. 33, No. 2, 870–918

Ng-Loy Wee Loon, Legitimizing Reverse Engineering Of Computer Programs In Copyright Law – How Far Have We Gone In Singapore?, Int. J. Law Inf. Technol. 1996, Vol. 4, No. 1, 48–64

NGUYEN NHAN/NADI SARAH, An Empirical Evaluation of GitHub Copilot's Code Suggestions, 2022 IEEE/ACM 19th International Conference on Mining Software Repositories (MSR), Pittsburgh, PA, USA 2022, 1–5

NGUYEN-DUC ANH/CABRERO-DANIEL BEATRIZ/PRZYBYLEK ADAM/ARORA CHETAN/KHANNA DRON/HERDA TOMAS/RAFIQ USMAN/MELEGATI JORGE/GUERRA EDUARDO/KEMELL KAI-KRISTIAN/SAARI MIKA/ZHANG ZHEYING/LE HUY/QUAN THO/ABRAHAMSSON PEKKA, Generative Artificial Intelligence for Software Engineering – A Research Agenda, 28 October 2023, available at: https://arxiv.org/abs/2310.18648 (last accessed: 19 July 2024), (cit.: Nguyen-Duc et al.)

ONGSULEE PARIWAT, Artificial Intelligence, Machine Learning and Deep Learning, Fifteenth International Conference on ICT and Knowledge Engineering 2017

PAGE LAWRENCE, Method for node ranking in a linked database, US 6285999, 4 September 2001 (cit.: US 6285999)

PAPASTEFANOU STEFAN, Smart Grids and Machine Learning in Chinese and Western Intellectual Property Law, IIC 2021, Vol. 52, No. 8, 989–1019

PATON MARK/MORTON JEREMY, Copyright Protection for Software Written By Software, Another look at the English law position, CRi 2011, Vol. 12, No. 1, 8–13

PENG SIDA/KALLIAMVAKOU EIRINI/CIHON PETER/DEMIRER MERT, The Impact of AI on Developer Productivity: Evidence from Github Copilot, 13 February 2023, available at: https://arxiv.org/abs/2302.06590 (last accessed: 19 July 2024), (cit.: PENG et al.)

PFEIFFER AXEL, Softwareprobleme im Patentrecht? Forschungs- und Entwicklungsprojekte, Informatik Forsch. Entw. 2004, Vol. 19, No.1, 30–40

PICHT PETER GEORG, Vom materiellen Wert des Immateriellen, Habil., Zurich/Munich 2018

PICHT PETER GEORG/BRUNNER VALERIE/SCHMID RENA, Artificial Intelligence and Intellectual Property Law: From Diagnosis to Action, Max Planck Institute for Innovation & Competition Research Paper No. 22-08, 2022 (cit.: PICHT et al.)

POLDRACK RUSSELL A./LU THOMAS/BEGUŠ GAŠPER, AI-Assisted Coding: Experiments with GPT-4, 25 April 2023, available at: https://arxiv.org/abs/2304.13187 (last accessed: 19 July 2024), (cit.: POLDRACK et al.)

POSNER RICHARD A., Economic Analysis of Law, 9th edn., New York 2014

POSNER RICHARD A., Intellectual Property: The Law and Economics Approach, JEP 2005, Vol. 19, No. 2, 57–73

RAGOT SÉBASTIEN, WIGGER FABIAN, DAL MOLIN LUCA, LAPPERT NANDO, ANDRIJEVIC ANA, REINLE MICHAEL, GLARNER ANDREAS, MERZ JAMES, HANDLE MARCO, GOTTSCHALTK MATTHIAS, FISCHER BENNO, ANTHAMATTEN SYLVIA, CORDOBA ANAÏC, Copyright in artificially generated works, sic! 2019, No. 10, 573–579 (cit.: RAGOT et al.)

RAHMATIAN ANDREAS, Originality in UK Copyright Law: The Old "Skill and Labour" Doctrine Under Pressure, IIC 2013, Vol. 44, No. 1, 4–34

RAMALHO ANA, Intellectual Property Protection for AI-Generated Creations, Europe, The United States, Australia and Japan, Abingdon 2022

RAUBER GEORG, 3. Kapitel Computersoftware, in: Streuli-Youssef Magda (ed.), Urhebervertragsrecht, Zurich 2006

REDEKER HELMUT, IT-Recht, 8th edn., Munich 2023

REHBINDER MANFRED/HAAS LORENZ/UHLIG KAI-PETER, URG Kommentar, Urheberrechtsgesetz mit weiteren Erlassen und internationalen Abkommen, 4th edn., Zurich 2022 (cit.: REHBINDER et al., in: OFK)

REIMER DIETRICH/ULMER EUGEN, Die Reform der materiellrechtlichen Bestimmungen der Berner Übereinkunft, GRUR Int. 1967, Vol. 16, No. 12, 431–454

REN XIAOSHUAI, Tencent Dreamwriter, IIC 2020, Vol. 51, No. 5, 652–659

RICKETSON SAM/GINSBURG JANE CAROL, International Copyright and Neighbouring Rights, 3rd edn., Oxford 2022

RIGAMONTI CYRILL P., Urheberpersönlichkeitsrechte – Globalisierung und Dogmatik einer Rechtsfigur zwischen Urheber- und Persönlichkeitsrecht, Bern 2013

ROHNER TIM, Der Schutz von KI-Schöpfungen im schweizerischen Urheberrecht, ZGE 2019, Vol. 11, No. 1, 33–85

SAMUELSON PAMELA, A Framework for a New Legal Regime for the Protection of Software Innovation, Pat. & Licensing 1995, Vol. 25, No. 5, 23–29

SAMUELSON PAMELA/DAVIS RANDALL/KAPOR MITCHELL D./REICHMAN JEROME H., A Manifesto concerning the Legal Protection of Computer Programs, Colum. L. Rev. 1994, Vol. 94, No. 8, 2308–2431 (cit.: SAMUELSON et al., Colum. L. Rev. 1994)

SAMUELSON PAMELA/DAVIS RANDALL/KAPOR, MITCHELL D./REICHMAN JEROME H., A New View of Intellectual Property and Software, Comm. of the ACM 1996, Vol. 39, No. 3, 21–30 (cit.: SAMUELSON et al., Comm. of the ACM 1996)

SCHACK HAIMO, Urheber- und Urhebervertragsrecht, 10th edn., Tübingen 2021

SCHAWE NADINE, Blockchain und Smart Contracts in der Kreativwirtschaft – mehr Probleme als Lösungen?, MMR 2019, Vol. 22, No. 4, 218–223

SCHEFFLER DIETRICH, Monopolwirkung und Informationsfunktion von Patenten aus heutiger Sicht, GRUR 1989, Vol. 91, No. 11, 798–802

SCHMID JÖRG (ed.), Kommentar zum schweizerischen Zivilrecht, Obligationenrecht, Kommentar zur 1. und 2. Abteilung (Art. 1–529 OR), 4th edn., Zurich 2014 (cit.: AUTHOR, in: ZK)

SCHNEIDER ANNIKA L., Die Patentierbarkeit von Computerprogrammen, Eine grundlagenorientierte Untersuchung des Technizitätskriteriums unter Berücksichtigung ökonomischer, rechtspolitischer und rechtsvergleichender Aspekte, Diss., Cologne 2014

SCHNEIDER JOCHEN/SPINDLER GERALD, Der Kampf um die gebrauchte Software – Revolution im Urheberrecht?, CR 2012, Vol. 28, No. 8, 489–498

SCHÖNBERGER DANIEL, Deep Copyright: Up- and Downstream Questions Related to Artificial Intelligence (AI) and Machine Learning (ML), ZGE 2018, Vol. 10, No. 1, 35–58

SCHUMACHER PABLO, Schutzfähigkeit Künstlicher Intelligenz, Die immaterialgüterrechtliche Schutzfähigkeit von KI-Technologien, Diss., Berlin 2023

SCHWARZ CLAUDIA, Anpassungsbedarf bei der Rechtsprechung des EPA und der deutschen Spruchkörper zu computerimplementierten Erfindungen?, GRUR 2023, Vol. 125, No. 20, 1418–1426

SCHWARZ CLAUDIA/KRUSPIG SABINE, Computerimplementierte Erfindungen – Patentschutz von Software?, 2nd edn., Cologne 2018

SCHWEIZER MARK/ZECH HERBERT (eds.), Stämpflis Handkommentar zum Patentgesetz (PatG), Bern 2019 (cit.: AUTHOR, in: SHK)

SEEMANN MATTHIAS, Übertragbarkeit von Urheberpersönlichkeitsrechten, Diss., Bern 2008

SENFTLEBEN MARTIN, Grundprobleme des urheberrechtlichen Dreistufentests, GRUR Int. 2004, Vol. 53, No. 3, 200–211

SEVILLE CATHERINE, EU Intellectual Property Law and Policy, 2nd edn., Cheltenham/Northhampton 2016

SHAO YALI, AIPPI Group Report China Q244 – Inventorship of multinational Inventions, 7 June 2015, available at: https://aippi.soutron.net/Portal/Default/en-GB/Search/SimpleSearch (last accessed: 19 July 2024)

SHAPIRO CARL, Navigating the Patent Thicket: Cross Licenses, Patent Pools, and Standard Setting, in: Jaffe Adam B./Lerner Josh/Stern Scott (eds.), Innovation Policy and The Economy, London 2001, 119–150

SHEMTOV NOAM, A Study on inventorship in inventions involving AI activity, London 2019 (cit.: SHEMTOV, inventorship)

SHEMTOV NOAM, Beyond the Code: Protection of Non-Textual Features of Software, Oxford 2017 (cit. SHEMTOV, Non-Textual Features)

SINGER ROMUALD/STAUDER DIETER/LUGINBÜHL STEFAN (eds.), Europäisches Patentübereinkommen, Taschenkommentar, 8th edn., Cologne 2019 (cit.: AUTHOR, in: Singer et al.)

SLOWINSKI PETER R., Rethinking Software Protection, in: Lee Jyh-An/Hilty Reto/Liu Kung-Chung (eds.), Artificial Intelligence and Intellectual Property, Oxford 2021, 341–361

SMITH BRADFORD L./MANN SUSAN O., Innovation and Intellectual Property Protection in the Software Industry: An Emerging Role for Patents?, U. Chi. L. Rev. 2004, Vol. 71, No. 1, 241–264

SOMMERVILLE IAN, Software Engineering, 10th edn., Boston 2015

STIERLE MARTIN, Artificial Intelligence Designated as Inventor – An Analysis of the Recent EPO Case Law, GRUR Int. 2020, Vol. 69, No. 9, 918–924

STRAUB WOLFGANG, Immaterialgüterrechtlicher Schutz mit KI geschaffener Werke und Erfindungen, Jusletter 7. August 2023

STRAUB WOLFGANG, Informatikrecht, Einführung in Softwareschutz, Projektverträge und Haftung, Zollikon 2004 (cit.: STRAUB, Informatikrecht)

STRAUB WOLFGANG, Softwareschutz, Urheberrecht, Patentrecht, Open Source, 2nd edn.: Zurich/St. Gallen 2024 (cit.: STRAUB, Softwareschutz)

STRENC ALEXANDRU CRISTIAN, European Software Directives and European Software Patents, 2nd edn., Alphen aan den Rijn 2022

SUMIDA MASAYOSHI, Outline of Japanese Paten Law, Japan Patent Office, Asia – Pacific Industrial Property Center, Japan Institute for Promoting Invention and Innovation, Tōkai 2018, available at: https://www.jpo.go.jp/e/news/kokusai/developing/training/textbook/ (last accessed: 19 July 2024)

SZATTLER EDUARD, Patentability of Computer Programs, Masaryk Univ. J. Law Technol. 2007, Vol. 1, No. 1, 97–108

THAM WINNIE/JACOB SHEENA/JAP ARTHUR/XIANGHUA CAI/YANG EUGENE, Patentability of Computer Implemented Inventions, AIPPI Study Question 2017, available at: https://aippi.soutron.net/Portal/Default/en-GB/Search/SimpleSearch (last accessed: 19 July 2024), (cit.: THAM et al.)

THOUVENIN FLORENT, Irrtum: Je kleiner der Gestaltungsspielraum, desto eher sind die Schutzvoraussetzungen erfüllt, in: Berger Mathis/Macciacchini Sandro (eds.), Populäre Irrtümer im Urheberrecht Festschrift für Reto M. Hilty, Zurich 2008, 61–73

THOUVENIN FLORENT/PICHT PETER G., AI & IP: Empfehlungen für Rechtsetzung, Rechtsanwendung und Forschung zu den Herausforderungen an den Schnittstellen von Artificial Intelligence (AI) und Intellectual Property (IP), sic! 2023, No. 10, 507–523

TOWSE RUTH, Creativity, Incentive and Reward, An Economic Analysis of Copyright and Culture in the Information Age, Rotterdam 2000

TROLLER KAMEN, Grundzüge des schweizerischen Immaterialgüterrechts, 2nd edn., Basel 2005

TUR-SINAI OFER, Beyond Incentives: Expanding the Theoretical Framework for Patent Law Analysis, Akron L. Rev. 2012, Vol. 45, No. 1, 243–289

UENO TATSUHIRO, A General Clause on Copyright Limitations in Civil Law Countries: Recent Discussion on Japanese-Style Fair Use Clause, in: Balganesh Shyamkrishna/Ng-Loy Wee Loon/Sun Haochen (eds.), The Cambridge Handbook of Copyright Limitations and Exceptions, Cambridge 2021, 211–215 (cit.: UENO, A General Clause)

UENO TATSUHIRO, The Flexible Copyright Exception for 'Non-Enjoyment' Purposes – Recent Amendment in Japan and Its Implication, GRUR Int. 2021, Vol. 70, No. 2, 145–152

ULMER EUGEN, Urheber- und Verlagsrecht, 3rd edn., Berlin/Heidelberg/New York 1980

VALLONE VERA, Decision G 1/19 of the Enlarged Board of Appeal of the European Patent Office, sic! 2021, No. 9, 508–511

VALLONE VERA, Patentierung von computerimplementierten Simulationen, sic! 2019, No. 12, 659–674

VAUGHAN-NICHOLS STEVEN, Open source isn't ready for generative AI. How stakeholders are changing this light bulb together, ZD Net 2023, available at: https://www.zdnet.com/article/open-source-isnt-ready-for-generative-ai-how-stakeholders-are-changing-this-light-bulb-together/ (last accessed: 19 July 2024)

VIEREGGE HENRIKE, Gemeinsamer Standpunkt zur Richtlinie über die Patentierbarkeit computerimplementierter Erfindungen, GRUR 2005, Vol. 107, No. 55, 399–400

VIVANT MICHEL/BRUGUIÈRE JEAN-MICHEL, Droit d'auteur et droits voisins, 4th edn., Paris 2019

WANDTKE ARTUR-AXEL/BULLINGER WINFRIED (eds.), Praxiskommentar Urheberrecht, 6th edn., Munich 2022 (cit.: AUTHOR, in: Wandtke/Bullinger)

WANG HAN, Authorship of Artificial Intelligence-Generated Works and Possible System Improvement in China, Beijing L. Rev. 2023, Vol. 14, No. 2, 901–912

WEINZINGER ARNULF/SONN HELMUT, Das österreichische Gebrauchsmustergesetz, GRUR Int. 1995, Vol. 44, No. 10, 745–752

WIEBE ANDREAS/HEIDINGER ROMAN, Ende der Technizitätsdebatte zu programmbezogenen Lehren? - Anmerkungen zur EPA-Entscheidung "Auktionsverfahren/Hitachi", GRUR 2006, Vol. 108, No. 3 177–180

WILD GREGOR, Publizität und Register im gewerblichen Rechtsschutz, Basel 2022

WILDING FRANCES/BRICK THOMAS, How revised EPO guidelines affect treatment of AI inventions, IAM, 27 April 2022, available at: https://www.iam-media.com/article/how-revised-epo-guidelines-affect-treatment-of-ai-inventions (last accessed: 19 July 2024)

WU SCOTT, Introducing Devin, the first AI software engineer, Cognition AI Blog, 12 March 2024, available at: https://www.cognition.ai/blog/introducing-devin (last accessed: 19 July 2024)

YANISKY-RAVID SHLOMIT, Generating Rembrandt: Artificial Intelligence, Copyright, and Accountability in the 3A Era – The Human-like Authors are Already Here – A New Model, Mich. St. L. Rev. 2017, 659–726

YOSHIDA ETSUKO, Artificial Intelligence-related invention in Japan and its disclosure with a focus on human intervention, in: Mendoza-Caminade Alexandra (ed.), L'Entreprise et l'Intelligence Artificielle – les Réponses du Droit, Toulouse 2022, 225–247

YU ROBERT, The Machine Author: What Level of Copyright Protection Is Appropriate for Fully Independent Computer-Generated Works?, U. Pa. L. Rev. 2017, Vol. 165, No. 5, 1245–1270

ZECH HERBERT, Artificial Intelligence: Impact of Current Developments in IT on Intellectual Property, GRUR Int. 2019, Vol. 68, No. 12, 1145–1147

ZHANG CHENGUO, Introducing the Open Clause to improve copyright flexibility in cyberspace? Analysis and commentary on the proposed "two-step test" in the Third Amendment to the Copyright Law of the PRC, in comparison with the EU and the US, CL & SR 2017, Vol. 33, No. 1, 73–86

ZIRPOLI CHRISTOPHER, Generative Artificial Intelligence and Copyright Law, Congressional Research Service, 29 September 2023, 1–6, available at: https://crsreports.congress.gov/product/details?prodcode=LSB10922 (last accessed: 19 July 2024)

ZUBERBÜHLER IVO, Die Erschöpfung von Patentrechten, Eine Auslegung von Art. 9a PatG und Art. 27b LWG, Diss., Bern 2012

List of Materials

Botschaft des Bundesrates an die Bundesversammlung über drei Patentübereinkommen und die Änderung des Patentgesetzes vom 24. März 1976, BBl 1976 II 1 (cit.: BBl 1976 II 1)

Botschaft zu einem Bundesgesetz über das Urheberrecht und verwandte Schutzrechte (Urheberrechtsgesetz, URG), zu einem Bundesgesetz über den Schutz von Topographien von integrierten Schaltungen (Topographiengesetz, ToG) sowie zu einem Bundesbeschluss über verschiedene völkerrechtliche Verträge auf dem Gebiete des Urheberrechts und der verwandten Schutzrechte vom 19. Juni 1989, BBl 1989 III 477 (cit.: BBl 1989 III 477)

Botschaft zum Bundesbeschluss über die Genehmigung von zwei Abkommen der Weltorganisation für geistiges Eigentum und zur Änderung des Urheberrechtsgesetzes, BBl 2006 3389 (cit.: BBl 2006 3389)

Botschaft zur Änderung des Urheberrechtsgesetzes sowie zur Genehmigung zweier Abkommen der Weltorganisation für geistiges Eigentum und zu deren Umsetzung vom 22. November 2017 (cit.: BBl 2018 591)

Communication from the Commission to the European Parliament pursuant to the second subparagraph of Art. 251(2) of the EC Treaty concerning the common position of the Council on the adoption of a directive of the European Parliament and Council on the patentability of computer-implemented inventions, 9 March 2005, COM(2005) 83 final– 2002/0047 (COD)

Explanatory Memorandum, Proposal for a Directive of the European Parliament and of the Council on the patentability of computer-implemented inventions, 20 February 2002, C5-0082/02 – COM(2002) 92 final – 2002/0047 (COD)

National Commission on New Technological Uses of Copyrighted Works (CONTU), Final Report on the National Commission on New Technological Uses of Copyrighted Works, Computer L.J. 1981, Vol. 3 No. 3, 53–104 (cit.: CONTU Final Report)

Patentability of computer-implemented inventions European Parliament legislative resolution (***II) on the Council common position with a view to the adoption of a directive of the European Parliament and of the Council on the patentability of computer-implemented inventions, 6 July 2005, (P6_TA(2005)0275 – 11979/1/2004 – C6-0058/2005 – 2002/0047(COD))

Position (***I) of the European Parliament adopted at first reading on 24 September 2003 with a view to the adoption of Directive 2003/…./EC of the European Parliament and of the Council on the patentability of computer-implemented inventions, 24 September 2003, P5_TC1-COD(2002)0047

Procedure file 2002/0047(COD), Patent law: patentability of computer-implemented inventions, available at: https://oeil.secure.europarl.europa.eu/oeil/popups/ficheprocedure.do?lang=en&reference=2002/0047 (last accessed: 19 July 2024)

Proposal for a Directive of the European Parliament and of the Council on the patentability of computer-implemented inventions, 20 February 2002, (2002/C 151 E/05) – COM(2002) 92 final – 2002/0047(COD)

Report (***I) on the proposal for a directive of the European Parliament and of the Council on the patentability of computer-implemented inventions, 18 June 2003, A5-0238/2003, (COM(2002) 92 – C5-0082/2002 – 2002/0047(COD))

U.S. Congress, Office of Technology Assessment, Computer Software and intellectual Property – Background Paper, OTA-BP-CIT-61, Washington, DC 1990 (cit.: U.S. Congress, Computer Software)

UKIPO, Consultation outcome, A pro-innovation approach to AI regulation: government response, available at: https://www.gov.uk/government/consultations/ai-regulation-a-pro-innovation-approach-policy-proposals/outcome/a-pro-innovation-approach-to-ai-regulation-government-response (last accessed: 19 July 2024), (cit.: UKIPO, A pro-innovation approach to AI regulation 2024)

UKIPO, Consultation outcome, Artificial Intelligence and Intellectual Property: copyright and patents: Government response to consultation, available at: https://www.gov.uk/government/consultations/artificial-intelligence-and-ip-copyright-and-patents/outcome/artificial-intelligence-and-intellectual-property-copyright-and-patents-government-response-to-consultation (last accessed: 19 July 2024), (cit.: UKIPO, AI & IP)

List of Sources

Beijing Higher People's Court Opinions on Several Issues Relating to Patent Infringement Adjudication (for Trial Implementation), 29 September 2001, translation available at: http://www.chinantd.com/sources-page.asp?id=1691&stypeid=12 (last accessed: 19 July 2024), (cit.: Beijing Higher People's Court Opinions on Several Issues Relating to Patent Infringement Adjudication)

Cyberspace Administration of China, Office of the Central Cyberspace Affairs Commission, Interim Measures for the Management of Generative Artificial Intelligence Services, available at: https://www.cac.gov.cn/2023-07/13/c_1690898327029107.htm (last accessed: 19 July 2024)

Dreamwriter webpage, available at: https://dreamwriter.ai/ (last accessed: 19 July 2024)

EU, China IPR SME Helpdesk, How to File a Copyright Registration in China, available at: https://intellectual-property-helpdesk.ec.europa.eu/ip-management-and-resources/copyright_en (last accessed: 19 July 2024), (cit.: EU IP Helpdesk)

Examination Guidelines for Patent and Utility Model in Japan, April 2023, https://www.jpo.go.jp/e/system/laws/rule/guideline/patent/tukujitu_kijun/index.html (last accessed: 19 July 2024), (cit.: JPO Guidelines)

Examination Guidelines for Patent Applications at IPOS 2023, available at: https://www.ipos.gov.sg/docs/default-source/resources-library/patents/guidelines-and-useful-information/examination-guidelines-for-patent-applications.pdf (last accessed: 19 July 2024), (cit.: IPOS Guidelines)

Examination Handbook for Patent and Utility Model in Japan, March 2024, https://www.jpo.go.jp/e/system/laws/rule/guideline/patent/handbook_shinsa/index.html (last accessed: 19 July 2024), (cit.: JPO Handbook)

Guidelines for Examination in the European Patent Office, March 2023 edition, https://www.epo.org/law-practice/legal-texts/guidelines.html, (last accessed: 19 July 2024), (cit.: EPO Guidelines)

IGE, Richtlinien für die Sachprüfung der nationalen Patentanmeldungen sowie der ESZ- und PESZ-Gesuche, 1 July 2023, available at: https://www.ige.ch/de/etwas-schuetzen/patente/anmeldung-in-der-schweiz/sachpruefung (last accessed: 19 July 2024), (cit.: IGE Richtlinen)

IPOS Circular No. 2/2019: Launch of AI2: Accelerated Initiative for Artificial Intelligence: An Accelerated Application-to-Grant Service for Patent Applications in Artificial Intelligence, available at: https://www.ipos.gov.sg/about-ip/patents/practice-guidelines-circulars (last accessed: 19 July 2024), (cit.: IPOS Circular No. 2/2019)

IPOS Circular No. 3/2022, Extension of SG IP FAST Track Programme and Sunset of 12 Months File-to-Grant Programme, available at: https://www.ipos.gov.sg/about-ip/patents/practice-guidelines-circulars (last accessed: 19 July 2024), (cit.: IPOS Circular No. 3/2022)

JPO, Comparative Research on the Patent Systems of Japan, the United States and Europe, Asia-Pacific Industrial Property Center, Japan Institute for Promoting Invention and Innovation (2017), available at: https://www.jpo.go.jp/e/news/kokusai/developing/training/textbook/ (last accessed: 19 July 2024), (cit.: JPO, Comparative Research Japan – US – Europe)

Patent Examination Guidelines 2023 (CNIPA Office Order No. 78), China, available at: https://www.wipo.int/wipolex/en/legislation/details/22371 (last accessed: 19 July 2024), (cit.: CNIPA Guidelines)

SOFTIC, Introduction to Computer Program Registration, available at: https://www.softic.or.jp/index.php/en/services/program-registration (last accessed: 19 July 2024)

SOFTIC, Statistics on Applications for Program Registration, available at: https://www.softic.or.jp/index.php/en (last accessed: 19 July 2024)

Stephen Thaler v. Vidal, Petition for Writ of Certiorari to the United States Court of Appeals for the Federal Circuit, Brief of Amici Curiae Brooklyn Law Incubator & Policy (BLIP) Clinic and Prof. Dr. Peter Georg Picht in support of petitioner, No. 22-919, https://www.supremecourt.gov/search.aspx?filename=/docket/docketfiles/html/public/22-919.html (last accessed: 19 July 2024), (cit.: *Thaler v. Vidal*, Brief of Amici Curiae)

Stephen Thaler v. Vidal, Petition for Writ of Certiorari to the United States Court of Appeals for the Federal Circuit, available at: https://www.supremecourt.gov/search.aspx?filename=/docket/docketfiles/html/public/22-919.html (last accessed: 19 July 2024)

The Artificial Inventor Project, court proceedings regarding patents, available at: https://artificialinventor.com/patent/ (last accessed: 19 July 2024)

The Yomiuri Shimbun, Tokyo District Court Rules AI Cannot Be Issued Patents; Law Recognizes Only 'Natural Persons' as Inventors, 17 May 2024, available at: https://japannews.yomiuri.co.jp/society/crime-courts/20240517-186568/ (last accessed: 19 July 2024)

UKIPO, Application No. BL O/741/19, available at: https://www.ipo.gov.uk/p-challenge-decision-results/p-challenge-decision-results-bl?BL_Number=O/741/19 (last accessed: 19 July 2024).

UKIPO, Manual of Patent Practice (Updated April 2024), available at: https://www.gov.uk/guidance/manual-of-patent-practice-mopp (last accessed: 19 July 2024)

USCO, Cancellation Decision of 21 February 2023 re: Zarya of the Dawn (VAu001480196), available at: https://www.copyright.gov/docs/zarya-of-the-dawn.pdf (last accessed: 19 July 2024), (cit.: USCO, Zarya of the Dawn)

USCO, Circular 1, Copyright Basics, available at: https://www.copyright.gov/circs/ (last accessed: 19 July 2024), (cit.: USCO, Circular 1)

USCO, Circular 61, Copyright Registration of Computer Programs, https://www.copyright.gov/circs/ (last accessed: 19 July 2024), (cit.: USCO, Circular 61)

USCO, Compendium of U.S. Copyright Office Practices, 3rd edn., 2021, available at: https://www.copyright.gov/comp3/ (last accessed: 19 July 2024), (cit.: USCO, Compendium)

USCO, Copyright Registration Guidance: Works Containing Material Generated by Artificial Intelligence, available at: https://www.copyright.gov/ai/ (last accessed: 19 July 2024), (cit.: USCO, Copyright Registration Guidance)

USPTO, 2019 Revised Patent Subject Matter Eligibility Guidance, Document No. 2018-28282, available at: https://www.federalregister.gov/documents/2019/01/07/2018-28282/2019-revised-patent-subject-matter-eligibility-guidance (last accessed: 19 July 2024), (cit.: USPTO, 2019 Revised Patent Subject Matter Eligibility Guidance)

USPTO, Application No. 16/524,350, available at: https://www.uspto.gov/sites/default/files/documents/16524350_22apr2020.pdf (last accessed: 19 July 2024)

USPTO, Inventorship Guidance for AI-assisted inventions, Federal Register, Vol. 89, No. 30, 13 February 2024, available at: https://www.govinfo.gov/content/pkg/FR-2024-02-13/pdf/2024-02623.pdf (last accessed: 19 July 2024), (cit.: USPTO, Inventorship Guidance for AI-assisted inventions)

USPTO, Manual of Patent Examining Procedure, 9th Edition, 2023, available at: https://www.uspto.gov/web/offices/pac/mpep/index.html (last accessed: 19 July 2024)

USPTO, Press Release of 12 February 2024, available at: https://www.uspto.gov/blog/director/entry/ai-and-inventorship-guidance-incentivizing (last accessed: 19 July 2024)

WIPO, Model Provisions on the Protection of Computer Software, Copyright – Monthly Review of the World Intellectual Property Organization (WIPO), No 1/1978, 6–16, available at: https://www.wipo.int/edocs/pubdocs/en/copyright/120/wipo_pub_120_1978_01.pdf (last accessed: 19 July 2024), (cit.: WIPO, Model Provisions)

Executive Summary

1 Copyright and patent protection for computer programs has evolved since the 1960s. Although it caters to the need for some sort of IP protection, this legal framework is far from ideal, as copyright and patent law were conceptualized for subject matter (literary and artistic works; technical inventions) that differs in many respects from computer programs. Despite these defects, the **market has learned to somehow cope** with the current software protection regime and its shortcomings. Nonetheless, almost **all experts and stakeholders** we talked to for this study **strongly support a reform** of the current system due to its shortcomings and the changes in market circumstances, notably the advent of generative Artificial Intelligence (AI). Indeed, the current system will face a fundamental challenge in the not-too-distant future, as computer programs are already and will increasingly be generated by AI systems. In jurisdictions which do not grant copyrights, or even patents in some cases, for machine-generated works or inventions, **AI-generated computer programs may remain unprotected** by the current IP regime.

2 Based on an assessment of Swiss law and the EPC, as well as on a comparative look at further important jurisdictions (elements of law in the EU, US, UK, Japan, China and Singapore), the study identifies the **main deficiencies** of the current software protection regime:

- Since many of its general rules and principles are a bad fit for computer programs, copyright law has developed a series of special provisions for computer programs which arguably amount to a specific regime ("Sonderurheberrecht").
- There is considerable uncertainty over the availability of patent protection for computer-implemented inventions; as the COMVIK approach becomes more nuanced and complex, it seems to disfavour MSMEs.[1]
- Transparency deficiencies regarding the software IP stack, the potential for protection thickets and harmful access restrictions, as well as resulting transaction costs render transactions over computer programs more cumbersome than desirable.
- Source code independently generated by AI lacks copyright protectability in many jurisdictions. Even patent protection may be rejected in some of them.

3 In view of these deficiencies, our study proposes a **Novel Approach** to IP protection of computer programs. However, it cannot comprehensively address all aspects of this approach, which will require **follow-on work, including interdisciplinary research and expert/stakeholder discussion**. The study does submit proposals to initiate and guide such follow-on work. In doing so, it starts from the assumption that **some form of IP protection should be available** also for computer programs that are partially or entirely generated by AI, not least because the software sector, including open source, relies heavily on such protection.

4 The fact that computer programs do not fit into either copyright or patent law suggests the creation of a new, **sui generis** IP right (the **"Software Right"**) for their protection. The international treaties currently in force do not restrict the introduction of such a new right, as they only harmonise the laws on existing IP rights.

5 The proposed Software Right would be two-tiered and protect the central added values of computer programs: a **Code-Level Software Right ("Code SR")** would protect the concrete formulation of the source code and its translation into binary code. A **Functionality-Level Software Right ("Functionality SR")** would protect concrete ways (modalities) in which computer programs

[1] Micro, small and medium-sized enterprises.

perform functionalities. While the two rights would protect different aspects of software and therefore differ in terms of subject matter and protection requirements, they would be largely parallel in terms of the rights granted, their limitations and the term of protection.

6 The Software Right would meet the following **market and user needs** inter alia:

- automatic protection of the source code upon generation; availability of functionality-based protection
- severance from the human author/inventor principle, which would allow for the protection of software irrespective of whether or not it was developed by means of AI
- allocation and ownership rules adequate for market realities, such as (default) corporate ownership
- transparency through registration, with an effective registration system based on a digital register
- comprehensive economic rights, which would allow rights holders to control any use of the computer program, including private use, but with no moral rights
- full transferability of both the Code SR and the Functionality SR
- tailor-made limitations
- a non-excessive term of protection

7 The **protected subject matter** of the **Code SR** would be the source code in all its forms, including binary code. The **Functionality SR** would protect a specific solution to a specific problem. The subject matter of such a right would be neither an abstract goal nor an unlimited number of potential solutions to a problem. Instead, the applicant would have to claim one or several concrete ways ("modalities") in which a computer program, running on a machine, performs the functionality, and the protection would only extend to these modalities. Consequently, the application for a Functionality SR would have to contain code demonstrating this and how the claimed modality(-ies) can be implemented. An appropriate tailoring of the protection requirement and of limitations should mitigate the risk of overprotection associated with the protection of functionalities.

8 Regarding the **requirements for protection**, a **Code SR** arises if the software developing entity (including AI) uses the scope, which is potentially limited but still available, to take creative and/or innovative code-writing decisions in such a way that the probability of an independent creation of (virtually) identical source code appears to be very small. This assessment should be made from the perspective of an expert taking into account standard coding practice. As with other registered IP rights, the **Functionality SR** would only be granted for **novel** modalities which would be assessed against the state of the art at the time of application. In addition, an innovative step of sufficient quality beyond the state of the art is required, formulated as a **non-obviousness** requirement similar to that in patent law. A Functionality SR should only be granted if the applicant can show that the modality can be **executed on a machine**, namely by implementing it in source code.

9 The **Code SR** should arise with the **creation of the code**, with protection taking effect as soon as the code meets the requirements for protection. There should be no prior examination and no formalities. It should be possible, but not mandatory, to register the source code in a software register ("SR Ledger"). The acquisition and registration of the **Functionality SR** would require an application, an assessment of the requirements for protection by an IP office and a registration. The **natural person** who wrote the code or developed the modality, or the **legal person** that organised and financed these activities, should be **entitled to the Code SR or, respectively, the right to apply for a Functionality SR**. If an AI system independently generated code and/or modalities, the user

of the system should, by default, acquire the rights to such output.[2] Both SRs would be **fully transferable and licensable** to other legal or natural persons. Transfers of and exclusive licences to Functionality SRs would have to be registered in the **SR Ledger**. The SR Ledger should allow for smart contracting features.

10 Both SRs should grant comprehensive **economic rights** covering any commercial or private use of the protected subject matter, but no moral rights. The **scope of protection** of the Code SR would encompass all uses of identical and sufficiently similar code, including the mere translation into a different programming language. The Functionality SR would cover the claimed modality(-ies) in any source code and use on any machine.

11 Both SRs should be subject to a series of **limitations**. The protection of functionalities, in particular, can lead to relatively far-reaching restrictions on the market or innovation activities of third parties. The Functionality SR must therefore be restricted by appropriate limitations. Although the risk of impeding innovation is smaller with the Code SR, limitations are equally important to ensure that this right does not restrict future code development, interoperability with other programs, and a seamless and secure use of the program. While many limitations would be more relevant for the Code SR or the Functionality SR, only some limitations will be clearly irrelevant for one of the two rights, such as limitations for the development of dependent programs or in case of lack of use. We therefore propose, in principle, a uniform regime of limitations for both types of SRs to allow for the use of the protected subject matter for scientific research, error correction (bug fixing), decompilation, backup copies, security purposes, and the training, validation and testing of AI systems. Further research and discussion should evaluate the need for additional limitations, including protection of competition and dynamic efficiency.

12 **Term of Protection:** We propose an **initial protection period of five years** for both the Code SR and the Functionality SR. This initial period should be **renewable**, subject to the payment of a (progressive) renewal fee, and subject to registration regarding the Code SR. The maximum term of protection after renewal should be no longer than **15 years**.

13 In addition to the long-term project of developing a new sui generis Software Right, the **current IP regimes should be revised** to address their shortcomings. The scope of such adaptations is, however, limited by the given framework of international law, in particular Art. 10(1) TRIPS Agreement and the provisions of the Berne Convention.

14 Nonetheless, important **improvements** to current **copyright protection** are possible. We suggest the following:

- appropriately adapting requirements for protection
- allowing for initial ownership of copyrights in computer programs by legal persons
- establishing a software register which allows the owner of a copyright in a computer program to register the source code on a ledger
- jettisoning moral rights in computer programs
- establishing a full-fledged, computer-program-specific catalogue of limitations, including for error correction (i.e. bug fixing) decompilation, backup copies, analysing and adapting computer programs for security purposes, use of computer programs for scientific purposes, use of computer programs for training, validating, and testing AI systems

[2] Contractual arrangements on the transfer of rightsholdership remain possible. An exceptions rule may have to add flexibility to the above default rule, but this should be further assessed by research and expert discussion.

- reducing the term of protection to a period of 50 years after commercialization of a computer program or, failing such commercialization, 50 years after the development of the program

15 These provisions may be implemented within the framework of existing copyright provisions. Instead, legislators could create a **specific section on computer programs** in their copyright acts, thereby acknowledging that a specific regime ("Sonderurheberrecht") for computer programs has already evolved over the years.

16 With regard to **patent law**, we suggest abandoning the current case law path of gradually, and not always coherently, relaxing the technical character requirement. Instead, we advocate a bolder approach by accepting software-integrating patent claims as long as they describe a concrete performance of a **functionality of a computer program**. This would require abandoning parts of the COMVIK approach and adapting patent examination guidelines. A modification of Art. 52(2)(c) EPC may facilitate the shift but does not seem inevitable, since the "as such" wording could be understood to merely exclude software that is not part of a "functionality" claim.

17 In addition, **patent laws** should allow for the initial acquisition of patents by legal persons **("corporate patents")** and for the **designation of AI systems as inventors**, including information on the nature and scope of the system's inventive activity, so as to better reflect today's innovation processes and to allow for a seamless integration of AI-generated inventions.

18 Going forward, we envision a **three-stage implementation scenario** for the Novel Approach:

- Stage 1: **Improvements** to the current legal framework and **further research/discussion** on further developing a sui generis Software Right;
- Stage 2: **Implementation** of a sui generis Software Right, preferably through an EPC-style international treaty but possibly also in individual, pioneering jurisdictions that wish to improve their IP framework for computer programs; and **coexistence** of the sui generis Software Right within the existing IP law framework. During such a coexistence phase, markets would assumingly gravitate towards using the sui generis right, aided by a restrictive practice regarding the patent and copyright protectability of computer programs;
- Stage 3: **Predominant** use of the sui generis Software Right; possibility of explicit removal of software protection from patent and copyright law in sync with an adaptation of the TRIPS Agreement.

Novel Approach: sui generis Software Rights at a glance		
	Code SR	**Functionality SR**
Protected subject matter	Source code in all its forms	Modality: specific performance of a functionality by a computer program or software architecture running on a machine; possibly sector-specific
Protection requirement	Creative/innovative coding decision, considering limited leeway for creativity	Novelty and non-obviousness, successful performance on a machine
Acquisition	• Initial acquisition ipso jure upon code generation; no examination or formality requirement • (AI-based) examination and registration optional, registration benefits (see below)	• Application and SR Ledger registration • Full-fledged examination
Ownership	• Starting point: coder or modality developer • Corporate acquisition and ownership possible • AI systems: user default owner of output, contractual arrangements possible • Possibly exceptions (assessment of entitlement parameters) for special settings	
Effects of protection	• Any commercial or private use of protected subject matter, including reproduction and distribution • No moral rights • Identical and sufficiently similar code in scope	• Modalities as claimed, in any source code expression running on a machine • No moral rights • Obvious equivalences in scope
Transferability, licensing, SR Ledger	• Fully transferable, licensable; no non-transferable moral rights • Transfers of registered Software Rights under registration requirement • SR Ledger with strong (bona fide) protection for acquirers/licensees, including in cases of succession of ownership • SR Ledger to provide smart contracting feature (e.g. automated standard licences)	
Minimum limitations (further differentiation between SR types subject to discourse)	• Protectability exclusion based on fundamental societal interests • Research privilege • Reverse engineering/decompilation for interoperability purposes • Bug fixing and IT security • Backup copy • Dependent programs • Lack of use • Training of AI systems, if not otherwise secured; possibility of compensation	
Term of protection (subject to discourse)	• 5 years initially • Extension possible; extension fee (increasing); registration requirement for Code SR extension • 15-year maximum protection period	

A. Introduction

19 This Study relates to **research undertaken by Zurich University's Center for Intellectual Property and Competition Law (CIPCO)**, in cooperation with inter alia the Swiss Federal Institute of Intellectual Property (IPI), in the field of **artificial intelligence and intellectual property law**. This research indicated that key issues at the intersection of AI and IP – issues such as non-human inventorship/authorship, collaborative content generation, a potential shift in the incentivisation rationale of IP law, and concerns over harm to innovation and competition from the massive AI generation of IP-protectable content – are **particularly virulent with regard to software.** Given the tasks it already performs and will increasingly perform in our societies and economies, **AI** further accentuates the role of computer code as the lingua franca backbone of a digitalising world. At the same time, as the present study will show, these developments risk to magnify the impact of flaws in the IP protection regime for software. Improvements to this regime would therefore make an important contribution to an appropriate IP law framework for the digital future.

20 To develop suggestions for such improvements, the study relied on **three main methods**: First, desk research clarified the state of law and discourse in the field. Second, a comparative assessment identified the approaches adopted by certain important jurisdictions other than Switzerland, including their evolving reactions to AI. Third, several exchanges with thought leaders from industry, bench and bar recorded experiences with the practical use of software IP rights, misgivings over the current shape of these rights, ideas for reform and, in a second and third round of exchanges, input on the suggestions made by this study. While we do not claim these exchanges are representative in the statistical sense of the word, we carefully selected experts from the most important stakeholder groups: large and small software developing companies, companies more on the in-licensing side, legal practitioners representing these various company types, high-ranking representatives of Swiss and other IP offices with a purview including software protection, and leading academics with a focus on software copyright.

21 The scope of the study **focuses on patent and copyright law**. Not only does this keep the study concise, it also reflects the more pressing need for reform in patent and copyright law, as well as the perception that general trade secrets/unfair competition rules will, in any case, continue to cover software and "be around" as an accompanying, and sometimes fallback, protection regime. Future research by the authors of this study will assess in more detail the interplay of these elements with a recast software IP rights regime.

B. Software Development

I. Software

22 A **broad definition** of the term "software" encompasses computer programs, related configuration data and necessary documentation.[3] For the purposes of this study, a narrower understanding seems more workable, namely a focus on computer programs. Therefore in this study we use the terms "software" and "computer programs" interchangeably.

23 In its "Model Provisions on the Protection of Computer Software", the **WIPO defined** computer programs as "a set of instructions capable, when incorporated in a machine-readable

[3] SOMMERVILLE, 19; SCHNEIDER, ch. 1 para. 24; SCHWARZ/KRUSPIG, ch. 2 para. 73; see LEINS-ZURMUEHLE, para. 123.

medium, of causing a machine having information processing capabilities to indicate, perform or achieve a particular function, task or result".[4]

24 This definition is, in principle,[5] broad enough to capture computer programs that qualify as Artificial Intelligence (AI). **According to Art. 3(1) of the EU AI Act,[6] an AI system is** "[…] a machine-based system designed to operate with varying levels of autonomy and that may exhibit adaptiveness after deployment, and that, for explicit or implicit objectives, infers, from the input it receives, how to generate outputs such as predictions, content, recommendations, or decisions that can influence physical or virtual environments". As opposed to traditional software, some AI systems, namely the ones that are based on machine learning, are not programmed in a traditional sense but are trained on large amounts of data. In addition, some AI systems are not deterministic, i.e. the same input does not necessarily produce the same output (see below).

II. Algorithms

25 Computer programs are based on algorithms.[7] An algorithm is a **procedure that is used to solve a specific task**. It consists of a sequence of step-by-step instructions or actions with which the task is solved in a specific way.[8] Algorithms are automatable instructions that process input data into output data according to a predefined process.[9]

26 A distinction must be made between **deterministic and non-deterministic algorithms**. Deterministic algorithms are static and follow a clearly defined solution path, even if they process a potentially unlimited number of values.[10] The entire process is clearly defined, and the same input always provides the same output. Non-deterministic algorithms, on the other hand, contain non-predictable components, e.g. the training data of a machine-learning system, which determine how the system later decides. Non-deterministic algorithms include, in particular, "learning" algorithms that are dynamic and can evolve.[11] These algorithms are therefore not limited to the commands in the program code but evolve through the influence of non-predictable elements.

III. Traditional Software Development

27 Traditional software development consists of **several stages**,[12] namely project planning, problem analysis, software design, implementation in software code, software testing and integration,

---

4 WIPO, Model Provisions, Sec. 1.

5 Some scholars deny that AI-related software generally falls under the WIPO definition; see for example SCHUMACHER, 158 et seq. on certain forms of neural networks.

6 Regulation (EU) 2024/[…] of the European Parliament and of the Council of […] laying down harmonised rules on artificial intelligence and amending Regulations (EC) No 300/2008, (EU) No 167/2013, (EU) No 168/2013, (EU) 2018/858, (EU) 2018/1139 and (EU) 2019/2144 and Directives 2014/90/EU, (EU) 2016/797 and (EU) 2020/1828 (Artificial Intelligence Act); final version available at: https://www.europarl.europa.eu/doceo/document/TA-9-2024-0138-FNL-COR01_EN.pdf (last accessed: 19 July 2024).

7 SCHWARZ/KRUSPIG, ch. 2 para. 61.

8 MARTINI, 17.

9 PAPASTEFANOU, IIC 2021, 993; LEINS-ZURMUEHLE, para. 124; KMENT/BORCHERT, ch. 2 para. 6.

10 MARTINI, 19; DORNIS, KI und Design, para. 5.

11 MARTINI, 19 et seq.; DORNIS, KI und Design, para. 6; ERTEL, 22.

12 BARENKAMP et al., AI Perspectives & Advances 2020, 3; LEINS-ZURMUEHLE, para. 158; SOMMERVILLE, 47.

as well as software support and maintenance.[13] The detailed concept specifies the structural layout of the computer program and how it will solve the posed task.[14]

28 The concept for the computer program is implemented in a programming language (Java, Python, etc.) as the **source code**. This does not require programmers to hand-type every line of code. For a broad variety of functions, suitable program components are available in software libraries. These components are often combined to develop a computer program or integrate it into existing code, without the need to write the entire code from scratch.[15] However, the software development process usually includes improvements ("refactoring") to such existing components, e.g. reductions in code complexity.[16]

29 Once the source code is complete, compiler programs convert it into the **object code**. The object code is machine readable and consists solely of binary signals.[17] The binary code is subsequently saved as an executable file (i.e. an ".exe" file) on a storage medium and can now be run on a computer. Running the program may require installation on a personal device (e.g. a laptop or a smartphone) or access and operation via a server, such as Microsoft 365.[18]

30 Subsequent to implementation, the testing phase examines through **test runs and error checks** whether the software works as intended.[19] The process of testing, bug fixing and retesting often takes up to 50% of the development time.[20]

31 **Software maintenance** concerns the process of changing a system after its delivery. In this process the software is constantly adapted to accommodate new requirements and to correct coding errors.[21]

IV. AI-Supported Software Development

32 With the rapid development of more computational power and neural networks that are now able to process vast amounts of data in a meaningful way, a variety of **new types of software products have emerged**.[22] Besides the large language models that are able to perform a plethora of tasks (e.g. ChatGPT)[23] or that can create artistic images from a prompt (e.g. Dall E·2),[24] there are now sophisticated AI systems that can be used at every stage of the software development process.[25] For example, certain AI systems automatically detect mismatches between the program specification and

---

13 BARENKAMP et al., AI Perspectives & Advances 2020, 3.

14 SOMMERVILLE, 197.

15 STRAUB, Softwareschutz, paras. 7 et seqq.

16 SOMMERVILLE, 278 et seq.; FOWLER, 85 et seqq.

17 RAUBER, 131; STRAUB, Softwareschutz, paras. 10 et seqq.

18 See STRAUB, Softwareschutz, paras. 10 et seqq.

19 BATARSEH et al., 199; SOMMERVILLE, 227 et seqq.

20 BATARSEH et al., 199.

21 SOMMERVILLE, 270 et seqq.

22 KREUTZER/SIRRENBERG, 74 et seqq.

23 Available at: https://chat.openai.com/ (last accessed: 19 July 2024).

24 Available at: https://openai.com/dall-e-2 (last accessed: 19 July 2024).

25 BARENKAMP et al., AI Perspectives & Advances 2020, 3; HARMAN, 3 et seq.

its implementation, helping to correct source code errors.[26] Other tools repeatedly execute programs on artificial data and collect, store and interpret the test results.[27] Furthermore, AI systems can automatically refactor software components[28] or generate code based on natural language input.[29]

33 As these examples show, AI arguably cannot yet – at least not on an industrial level – create (complex) computer programs entirely on its own, but **rather carries out certain steps in the development process**, such as refactoring software components or generating code based on natural language input.[30] At the same time, the systems already display a **remarkable proficiency**; they solve complex programming problems and write the corresponding code autonomously. In December 2021, Deepmind's AlphaCode[31] was tested in ten coding competitions with over 5,000 participants each,[32] where it was able to solve 29.6% of the programming problems. In doing so, the AI system achieved an average rank equal to the top 54.3% of human participants.[33] The system did not merely copy core logics from the training data; instead it adapted their underlying principles to the task at hand, similar to how a human programmer would proceed.[34] Therefore, based on a sufficiently detailed problem description, AI systems can solve novel programming problems at a performance level of a median human competitor.[35]

34 Another example is **Copilot**, jointly developed by GitHub[36] and OpenAI.[37] Prompted with a description in natural language of the target functionality, Copilot can generate suitable code in various programming languages.[38] Notwithstanding the fact that Copilot's solutions must be double-checked for errors and are often of limited complexity,[39] the tool substantially increases programmer productivity.[40] According to an experiment conducted by Peng et al., programmers using Copilot to solve a specific programming task were 55.8% faster than peers using conventional programming tools.[41] OpenAI's ChatGPT-4 is equally capable of generating code based on natural language input.[42]

---

26 LATINOVIC/PAMMER-SCHINDLER, 148.

27 BARENKAMP et al., AI Perspectives & Advances 2020, 7.

28 NGUYEN-DUC et al., 37 et seq.; POLDRACK et al., 3 et seqq.

29 NGUYEN/NADI, 1 et seqq.; PENG et al., 1 et seqq.; BUBECK et al., 21 et seqq.; POLDRACK et al., 1 et seqq.

30 LATINOVIĆ/PAMMER-SCHINDLER, 152.

31 AlphaCode was trained on 715 GB of human code that was taken from the platform GitHub; LI et al., Science 2022, 1093.

32 LI et al., Science 2022, 1094.

33 LI et al., Science 2020, 1092.

34 LI et al., Science 2020, 1095.

35 LI et al., Science 2020, 1096.

36 GitHub is a web-based platform offering a range of tools and services to support software development, https://github.com/ (last accessed: 19 July 2024).

37 OpenAI is an organization focused on AI research and development, https://openai.com/ (last accessed: 19 July 2024).

38 NGUYEN/NADI, 1; PENG et al., 2.

39 NGUYEN/NADI, 5.

40 PENG et al., 7.

41 PENG et al., 1 et seqq.

42 BUBECK et al., 21 et seqq.; POLDRACK et al., 6 et seq.

However, to ensure accuracy and minimize errors, the generated code must still be validated and improved by a human programmer, as with Copilot.[43]

35 More recent research evaluates AI systems in an environment that is closer to a real software engineering setting, the Software Engineering Bench (SWE-Bench),[44] in which the systems must resolve coding issues that are posted in popular GitHub repositories.[45] To achieve this, the systems are given an issue description and the respective flawed code.[46] The first experiments led to poor results; the systems were only able to solve the most simple issues, and the most successful system solved only 1.96% of all issues.[47] Moreover, the Python code written was relatively primitive, and the systems resolved the issues without having regard to code style or quality.[48] However, recent AI systems put to the same test had improved massively: Cognition's Devin[49] solved 13.86% of all issues and Factory's Code Droid[50] (the current market leader) solved 19.27% of all issues.[51] Considering that the initial study was submitted in October 2023[52] and that the subsequent quality increase was achieved in only seven months, it is clear that the systems are evolving rapidly.

36 These cases show that **problem/task identification and formulation remain with humans for the time being**. Humans are required for this key contribution, as AI systems do not themselves "think" of a problem to be solved.[53] When triggered by a generic prompt input, today's AI systems used in software engineering are not capable of autonomously generating (complex) computer programs. However, systems such as Copilot or ChatGPT-4 are capable of writing valuable source code, and constitute very useful tools for software engineers.

V. Autonomous Software Development by AI

37 Even though AI systems cannot yet autonomously develop (complex) computer programs, the impressive development of generative AI systems in recent years, the doubling of computing power every two years,[54] the ever-increasing amount of data, the large investments by Big Tech[55] and the significant efforts by many other software development companies, including start-ups, lead

---

44 JIMENEZ et al., ICLR Conference Paper 2024, 2.

45 JIMENEZ et al., ICLR Conference Paper 2024, 2.

46 JIMENEZ et al., ICLR Conference Paper 2024, 3.

47 JIMENEZ et al., ICLR Conference Paper 2024, 5.

48 JIMENEZ et al., ICLR Conference Paper 2024, 8.

49 Devin is an AI System said to be the first AI software engineer; see WU.

50 Similiarly to Cognition, Factory is building AI systems ("Droids") to automate software engineering, available at: https://www.factory.ai/ (last accessed: 19 July 2024).

51 The Leaderboard is available at: https://www.swebench.com/ (last accessed: 19 July 2024).

52 See date on arxiv page at: https://arxiv.org/abs/2310.06770 (last accessed: 19 July 2024).

53 BARENKAMP et al., AI Perspectives & Advances 2020, 6; HARMAN, 2.

54 KREUTZER/SIRRENBERG, 74.

55 See for example: https://www.forbes.com/sites/qai/2023/01/27/microsoft-confirms-its-10-billion-investment-into-chatgpt-changing-how-microsoft-competes-with-google-apple-and-other-tech-giants/?sh=699bc5a63624; https://www.nytimes.com/2023/01/23/business/microsoft-chatgpt-artificial-intelligence.html; https://www.aboutamazon.com/news/company-news/amazon-aws-anthropic-ai; https://inflection.ai/inflection-ai-announces-1-3-billion-of-funding (all last accessed: 19 July 2024).

us to believe that **AI systems will eventually, and maybe sooner than expected, be able to autonomously develop** computer programs.

38 Correspondingly, we expect that the role of humans will eventually be reduced to feeding a rather general task/problem to an AI system through prompts. In this scenario, **human input is reduced to such an extent that IP law will have to address the legal treatment of software without a human creator**. Proposed solutions for this stage should start from a reflection on why IP law protects software at all.

C. Rationales for Software Protection

39 The awarding of IP rights by legislators constitutes a **market intervention**, conferring exclusive rights to the rights holders. These rights allow the rights holders to control the use of a (public) good and to exclude third parties from using it. Such an exclusion is **only justified if the benefits of granting such rights outweigh the associated disadvantages**. Therefore, the granting of IP rights always requires a thorough assessment of their economic impact to ensure that their implementation is justified. This also applies to the granting of patents and copyrights for computer programs. While the rationales for patent protection have been analysed and discussed in considerable detail in the economic and legal literature, the rationales for granting copyright protection have received much less attention. This applies both to the general discussion of the rationales and to the more specific discussion of whether IP rights should be granted for computer programs.

I. Patent Law

1. Incentive Theory

40 The most important rationale for granting patents is the incentive theory: the patent system, by granting exclusive rights, provides **incentives to invest in research and development, thereby stimulating innovation**.[56] As patents also cause social costs by excluding others from using (public) goods, patent protection is unconvincing where deemed unnecessary for fostering innovation.[57]

41 In applying the incentive theory to computer programs, some argue that the software industry demonstrated innovation and productivity prior to the availability of patent protection for computer programs, which calls into question the rationale for such protection.[58] The **impact of patent protection on innovation is perceived to be less significant in the case of subject matter whose main value lies with computer programs**. This is because developers may succeed in appropriating the returns of their innovation even in the absence of a property right (e.g. rough effective price differentiation), where the price of the software license depends on the number of users, or by capitalizing on a "first-mover advantage".[59]

42 Moreover, with regard to **sequential and cumulative innovation**, as is the case for software development as such, it is asserted that patent protection may not promote innovation but rather

---

[56] MAZZOLENI/NELSON, J. Econ. Issues 1998, 1034 et seqq.; EVANS/LAYNE-FARRAR, Va. J.L. & Tech. 2004, para. 44.

[57] MAZZOLENI/NELSON, Res. Policy 1998, 275.

[58] EVANS/LAYNE-FARRAR, Va. J.L. & Tech. 2004, paras. 46 et seqq.; BESSEN/MASKIN, RAND J. of Econ. 2004, 611 et seq.; BESSEN/HUNT, 162.

[59] BESEN/RASKIND, J. Econ. Perspect. 1991, 5; BESSEN/MASKIN, RAND J. of Econ. 2004, 612.

impede it.[60] Software innovation is characterized as sequential because it frequently relies on pre-existing inventions within the sector. Furthermore, the innovation is often cumulative, as computer programs are frequently not protected by a single patent but consist of a combination of various components, each subject to one or more patents.[61] These attributes can result in a proliferation of software patents, which need to be considered before commercializing a product. As a result, extensive patent searches and multiple licences are required to avoid patent infringement, escalating development costs and introducing potential legal risks.[62] Given these considerations, some argue that granting patents for computer programs "as such" may stifle innovation, thereby undermining the validity of the incentive theory.[63]

2. Development and Commercialization of Inventions

43 The theory that patents are needed to further develop and commercialize inventions emphasizes the importance of patent protection to foster the development of inventions **until they can be commercialized as market-ready products**.[64] This theory suggests that patents provide assurance that, if development is successful, the economic rewards can be appropriated by preventing third parties from imitating the product. Consequently, patent protection encourages a favourable decision to invest in further development of the invention, thereby ensuring that inventions are actually developed into marketable products.[65] This reasoning is especially convincing in situations where the cost for making an invention is relatively small compared to the cost of developing a marketable product.

44 This rationale is closely related to the incentive theory but differs in focus, which is not on the making of the initial invention but on facilitating its commercialization. Since **computer programs are expensive to develop but inexpensive to imitate, this theory seems particularly well-suited to explain the granting of patent protection for computer program**s.[66] This is all the more so in the context of micro, small and medium-sized enterprises (MSMEs), including start-ups, which depend on funding for the development of a marketable product. The patent provides important assurance for prospective investors, thereby facilitating the raising of capital.[67]

3. Disclosure Inducement

45 A key justification for the patent system is its ability to promote disclosure of inventions: the disclosure of the invention is required for the granting of a patent, and it fosters dissemination of technical knowledge. Consequently, the patent system **provides important incentives against keeping inventions secret by offering an exclusive right in return for their disclosure**. Secrecy would often have adverse welfare consequences, since owners of the secret are rarely in a position to

---

60 EVANS/LAYNE-FARRAR, Va. J.L. & Tech. 2004, paras. 52 et seq.; BESSEN/MASKIN, RAND J. of Econ. 2004, 613; JAFFE, Res. Policy 2000, 552 et seq.; LANDES/POSNER, 408.

61 LANDES/POSNER, 408; BESSEN/MASKIN, RAND J. of Econ. 2004, 612; BALLARDINI, SCRIPTed 2009, 208; SAMUELSON et al., Colum. L. Rev. 1994, 2346; EVANS/LAYNE-FARRAR, Va. J.L. & Tech. 2004, para. 53.

62 BALLARDINI, SCRIPTed 2009, 208; LEMLEY/SHAPIRO, Texas L. Rev. 1991, 1993; SHAPIRO, 119 et seq.; GARFINKEL et al., Issues Sci. Technol. 1991, 52.

63 BESSEN/MASKIN, RAND J. of Econ. 2004, 612; SHAPIRO, 119 et seq.

64 MAZZOLENI/NELSON, J. Econ. Issues 1998, 1040.

65 MAZZOLENI/NELSON, J. Econ. Issues 1998, 1040; GRUNER, St. John's L. Rev. 2000, 1012.

66 HECKEL, Comm. of the ACM 1992, 124; GRUNER, St. John's L. Rev. 2000, 1015.

67 GRUNER, St. John's L. Rev. 2000, 1015; see LERNER, RAND J. of Econ. 1994, 319.

exploit or even recognize all potential applications of their invention.[68] This general legitimization is also applied with regard to software.[69]

46 In the absence of patent protection, it seems likely that enterprises seeking to protect their inventions will increasingly **resort to trade secrets** to protect their inventions, which may be more harmful to innovation than the existing system.[70]

47 On the other hand, given the highly **dynamic nature of the software industry**, some argue that information contained in a patent application filed years ago is often of very limited value.[71]

II. Copyright Law

48 As in patent law, the classical rationales for granting IP rights are also applied in copyright law. The **incentive theory** assumes that time and resources necessary for the creation of literary or artistic works are only invested if the income from the exploitation of the work allows for the amortization of the time and money invested in its creation, as well as for the reaping of a profit.[72] An approach similar to the theory that patent law induces the development of inventions **until their commercialization** can also be found in copyright law. In this respect, copyright law provides the rights holders with protection against unauthorized use of the work by third parties, without which they would likely not assume the substantial risks associated with the exploitation of the work. As a result, many works would remain unpublished.[73]

49 In copyright law, **natural law legitimation approaches** are of particular importance. For example, the labour theory is based on the idea that individuals have a natural right to control the product of their labour, including intellectual creations.[74] Similarly, the approach of personality protection denotes the ownership of authors as part of their personality over intellectual work they have created.[75] In the context of **computer programs, however, these rationales are rarely emphasized** (see also below, D.IV.1.c)).

50 The amendment of the Swiss Copyright Act of 1989 explicitly defined computer programs as works. The decisive rationale was the desire to achieve internationally harmonized rules for software protection.[76] Among the first countries enacting copyright protection for software was the United States, following a **recommendation from the Commission on New Technological Uses of Copyrighted Works** (CONTU).[77] This recommendation suggested that all uses of computer programs shall be protectable under copyright law.[78] The primary consideration was that since the cost of developing computer programs significantly exceeds the cost of their replication, these programs

---

68 LANDES/POSNER, J. Leg. Stud. 1989, 329; MAZZOLENI/NELSON, J. Econ. Issues 1998, 1039.

69 CAMPBELL-KELLY, Mich. Telecomm. & Tech. L. Rev. 2005, 198; GRUNER, St. John's L. Rev. 2000, 1007 et seqq.

70 EVANS/LAYNE-FARRAR, Va. J.L. & Tech. 2004, paras. 66 et seqq.; BALLARDINI, SCRIPTed 2009, 224.

71 JAFFE, Res. Policy 2000, 552.

72 DAVIES, para. 2.006; LANDES/POSNER, J. Leg. Stud. 1989, 346; TOWSE, 12; SCHACK, para. 5.

73 DAVIES, paras. 2.006, 9.008; MENELL, Stanford L. Rev. 1987, 148; TOWSE, 12.

74 MAY, 7; MENELL, Stanford L. Rev. 1987, 157 et seq.; HUBMANN, 31.

75 ULMER, 109 et seq.; TROLLER, 603; similar HUBMANN, 107; MENELL, Stanford L. Rev. 1987, 158 et seq.

76 BBl 1989 III 477, 501 et seq.

77 See CONTU Final Report.

78 CONTU Final Report, 54.

are likely to be disseminated only if the creator may recover its cost by way of some form of protection against unauthorized replication of the work. Consequently, the commission concluded that some sort of protection is essential to **incentivize the creation and commercialization of software**.[79] As in patent law, this reasoning is mainly based on the incentive theory as well as on the theory of fostering development until commercialization.[80] To accommodate the protection of computer programs in copyright law, the recommendation further argues that computer programs, like literary and artistic works, constitute a creation within the meaning of copyright law,[81] and it stresses that copyright law has historically demonstrated its capacity to adapt to new forms of expression.[82]

III. Findings

51 The brief analysis of key rationales for granting IP protection for computer programs has shown a justification strongly based on the **traditional rationales for patent and copyright protection**. A specific and **in-depth analysis of the economic implications** of patent and copyright protection for computer programs is **lacking in the literature**. In addition, the analysis is predominantly based only on theoretical considerations regarding the advantages and disadvantages of IP protection for computer programs. Although **some empirical research**[83] has been conducted regarding the impact of IP protection on the development of computer programs and innovation in the software industry, these studies are often not sufficiently conclusive.[84] Therefore, the arguments for and against granting IP protection for computer programs can, on the basis of existing empirical and economic research, **be neither fully confirmed nor rebutted**.

52 As so many of its predecessors, this study cannot definitively answer the question of whether and to what extent **current property-style IP rights are a *sine qua non*** for achieving the goals they have been stipulated for.[85] However, answering this question is not strictly necessary for a study which aims to develop a novel approach (hereinafter "**Novel Approach**") to software protection in a world (co)shaped by AI. This study does not analyse the question of *whether* IP rights should be granted for computer programs, but rather **focuses on *how* such IP rights should be designed**, taking into account the **increasing importance of AI-generated computer programs**. Of course, even with this focus the development of a Novel Approach to software protection must be based on the rationales for granting such protection, as these rationales provide important guidance regarding the design of such a Novel Approach.

53 Arguably, the most important reason for providing some form of IP protection for computer programs is that **today's IT industry heavily relies on existing IP protection** granted by

---

79 CONTU Final Report, 59.

80 See CONTU Final Report, 57.

81 CONTU Final Report, 59; see also MILLER, 982 et seqq.

82 U.S. Congress, Computer Software, 13 et seq.

83 A widely cited study found that, as a result of patent protection, American companies shifted their focus from developing new programs to filing patent applications and enforcing existing patents, which hinders the creation of new programs; see BESSEN/HUNT, 11 et seq., 14 et seq.

84 HILTY/GEIGER, IIC 2005, 630 et seqq.; see also BESSEN/HUNT, 16 emphasizing the need for further research.

85 Critical towards legislative intervention to foster innovation: HILTY et al., 71 et seq.; HILTY/GEIGER, IIC 2005, 630 et seqq., 646; Study that shows a negative impact of larger extent and duration of patents on innovation outcomes in the pharmaceutical industry: DOSI et al., J. Econ. Behav. Organ. 2023, 564 et seqq.; Study that finds a consistent and positive effect of IP on innovation: AMANKWAH-AMOAH/KEHINDE MEDASE, J. Knowl. Econ. 2023, 19 et seqq. See also BUDISH et al., Am. Econ. Rev. 2016, 183.

patent and copyright law. While it does seem possible to adapt the existing protection regime towards a framework that is both theoretically more convincing and capable of producing better results in practice (see below, F.II and F.III), doing away with IP protection for computer programs altogether seems virtually impossible given this condition of the IT industry. In addition, there is **no compelling evidence that a world without software IP rights would outperform** an improved version of the current system or, in particular, a recast set for software protection rights, **to the extent the heavy costs of a fundamental system change could be justified**.

54 In addition, certain of the above-mentioned rationales provide strong arguments for granting some form of IP protection for computer programs. Rewarding and thus **incentivizing investments** – especially **corporate** instead of individual investments – has always loomed relatively large for software compared to, say, notions of personality protection or natural law-enshrined compensation for personal labour.[86] Also, it is important to remember the various types of players in the software industry. While larger players would often find it easier to safeguard return on investment regardless of IP protection, **MSMEs active in software development** state clearly[87] that **IP rights are an important guarantor of their business model**. Intrinsic motivations to program and share code do exist,[88] but they do not outweigh commercial purposes in industrial software generation.

55 Furthermore, **transactability (licences, securitization, portfolio transfers, etc.) of software is key** to reaping its economic potential. IP helps to render software transactable as it allocates negotiable and appraisable rights to prospective transaction parties.[89] As one example for the manifold transactional benefits of rights allocation, **stakeholders reported** that demonstrable and quantifiable **software-rights portfolios are, or would be,[90] helpful in negotiating** (international) joint ventures or similar industry cooperations.

56 Finally, it was undisputed in our **stakeholder exchanges** that certain aspects of the existing system are beneficial and that a **complete abandonment of software protection would be detrimental**. This is particularly the case for the automatic granting of some form of protection, which is crucial for many **MSMEs**,[91] and for the **informational function of a registered right**, both in terms of providing transparency about existing IP rights and in terms of counteracting secrecy and thus enabling the efficient exploitation of an invention.[92]

57 These findings **also apply to Open Source Software (OSS)**, which promotes collaboration between developers worldwide so as to achieve faster development cycles and innovation

---

86 HILTY et al., 71; GRATTON, Comput. L. Rev. & Tech 2003, 251 et seq.; SMITH/MANN, U. Chi. L. Rev. 2004, 263. In general, on the IP law rationale of incentivizing investments by protection their results from being easily copied and, thus, devalued, see TUR-SINAI, Akron L. Rev. 2012, 248; POSNER, JEP 2005, 57 et seqq.; HILTY et al., 58 et seqq.

87 Inter alia in the input workshops leading up to this report.

88 DAPP, 138.

89 See PICHT, 9 et seq., 39 et seq., 54 et seq., with further references.

90 While patent register entries facilitate substantiating a party's software patent portfolio, registration-adverse copyright offers no such support.

91 HECKEL, Comm. of the ACM 1992, 124; JAFFE, Res. Policy 2000, 553; see also GRUNER, St. John's L. Rev. 2000, 1016 et seq.

92 WILD, 3 et seqq.; SCHEFFLER, GRUR 1989, 798 et seqq.; CAMPBELL-KELLY, Mich. Telecomm. & Tech. L. Rev. 2005, 198; GRUNER, St. John's L. Rev. 2000, 1007 et seqq.

through the exchange of ideas and resources.[93] The economic importance of OSS has risen continuously in recent years, and it is to be expected that its **relevance will continue to increase**.[94] OSS can be customised to meet specific needs. Companies have the freedom to change the code and adapt it to their requirements, resulting in more flexible solutions.[95] The source code of OSS can be viewed by anyone. This creates trust, as users and companies can examine the code, understand how a project works and verify its security and reliability.[96] A **large number of standard licences have been created for OSS** to meet the various wishes of the licensors, of which the MIT License and the GNU General Public License (GPL) are probably the most conspicuous examples.[97] These licences, and the associated functioning of OSS, are based on the IP (namely copyright) protection of software. **Without such protection**, third parties would not need to comply with the conditions of the applicable licence, and they **could further develop the code without sharing developments with the OSS community**. This would undermine the functioning of OSS, which has proven to be a very beneficial approach to software development, distribution and use in many sectors.

58 The advent of **software-generating AI systems** does not remove the rationales for granting IP rights for computer programs. The **most relevant rationales** – namely, the incentive theory, the disclosure theory, the need to grant IP rights to foster the further development of inventions to marketable products and the fostering of transactability, – **also apply in principle to AI-generated computer programs**. Arguably the most important difference between traditional code writing and AI-generated code is that the code is not written by human beings. Accordingly, **rationales based on the protection of the personality** of the author and the natural law-enshrined idea of a compensation for personal labour **do not hold** as rationales for the protection of AI-generated software. As mentioned above, however, these rationales have never played an important role for justifying the granting of IP rights for computer programs.

59 From an **economic perspective**, there are reflections on whether (software) **content generation will continue deserving incentivization through IP rights on the content output**, even though AI systems generate the content at irrelevant marginal cost.[98] Instead, the main investment worthy of protection and potentially in need of incentivization may take place at the level of producing content-generating AI systems. Even this perspective, however, would not justify jettisoning IP protection for AI-related computer programs.

D. Current Legal Situation

I. Preliminary Remarks

60 The development of a Novel Approach to the protection of computer programs in a world (co)shaped by AI must be based on a thorough analysis of the current approaches and frameworks in patent and copyright law. This analysis must consider both **"traditional" computer programs**

---

93 https://fastercapital.com/content/Harmony-in-Innovation--CCA-and-Open-Source-Software.html (last accessed: 19 July 2024).

94 HUPPERTZ, in: Leupold et al., § 2.4.2 para. 1.

95 GRÜTZMACHER, in: Wandtke/Bullinger, § 69c UrhG para. 109; REDEKER, paras. 97 et seqq.

96 https://fastercapital.com/content/Harmony-in-Innovation--CCA-and-Open-Source-Software.html (last accessed: 19 July 2024).

97 GRÜTZMACHER, in: Wandtke/Bullinger, § 69c UrhG para. 107; REDEKER, para. 98; see also the analysis of Black Duck Software, https://perma.cc/4DLA-88BH (last accessed: 19 July 2024).

98 See for example DE RASSENFOSSE et al., S. Cal. L. Rev. 2023, 105 seq.

and those generated by AI. It must take into account the framework provided by **international treaties**, especially the TRIPS Agreement (see below, D.II), while focusing on the concrete implementation of the IP protection for computer programs in **national (or regional) patent and copyright laws** (see below, D.III and D.IV)

61 As this study was commissioned by the Swiss IPI, its focus is on the legal situation in **Switzerland**. Other important jurisdictions, such as the **US, UK, Japan, China and Singapore**, are considered from a high-level comparative perspective. The main findings of this comparative analysis are integrated in the text of the study, while a separate **annex** contains details on respective approaches and regulatory landscapes.

II. International Treaties

62 A first decisive step in building the IP law framework for the protection of computer programs was the **European Patent Convention** of 1973.[99] **Economic considerations** were the primary reason for this solution; Europe lagged behind the US in the computer industry and there were concerns that the patentability of computer programs would further solidify this lead.[100] However, although the EPC excludes computer programs "as such" from patentability (Art. 52(2) i.c.w. Art. 52(3) EPC), the **"as such" wording in Art. 52(2) EPC was interpreted from the beginning to allow for the patent protection of software** in combination with technological inventions in the sense of Art. 52(1) EPC. Therefore, the EPO Guidelines were revised in 1985 to clarify that European patent law does not deny protection to new technical developments in computer programs.[101]

63 The protection of software via the **copyright system then took shape** as the **fastest way to achieve an effective protection regime at a global scale**.[102] The EU established copyright protection for software in 1991, through Art. 1 of the Council Directive on the legal protection of computer programs.[103] On a global level, member states of the WTO committed in the **TRIPS** Agreement to protecting computer programs (source and object code) as literary works in accordance with the **Berne Convention** (Art. 10(1) TRIPS). Art. 4 **WCT** confirms this by stipulating that copyright protection applies to computer programs, whatever their mode or form of expression.

64 International treaties therefore stipulate the protection of computer programs under **copyright law**. However, they do **not exclude other forms of software protection, especially through patent law**. Art. 27(1) TRIPS states that patents shall be available for any inventions, whether products or processes, and in all fields of technology, if they are new, involve an inventive step and are capable of industrial application. This wording is understood to permit the patenting of computer programs with their copyright protection.[104]

65 Against this international law background, the **EPO developed its approach to the protection of computer-implemented inventions** (see below, D.III.1 seq. and annex I.1), which

---

[99] HILTY/GEIGER, IIC 2005, 619.

[100] HILTY/GEIGER, IIC 2005, 619 fn. 15.

[101] ANN, § 12 para. 25; see EPO Guidelines, G-II, 3.6.

[102] STRAUB, Softwareschutz, para. 51.

[103] Council Directive 91/250/EEC of 14 May 1991 on the legal protection of computer programs, OJ L 122, 17 May 1991; later repealed by: Directive 2009/24/EC of the European Parliament and of the Council of 23 April 2009 on the legal protection of computer programs, OJ L 111, 5 May 2009.

[104] HEINRICH, in: PatG/EPÜ Kommentar, Art. 1 PatG/Arts. 52, 56, 57 EPÜ para. 40.

strongly influenced not only the EU member states but also Swiss law.[105] At the same time, both the **EU and Switzerland introduced copyright protection** for computer programs as requested by Art. 10(1) TRIPS. This dichotomy shapes the reality of today's software protection regime.

III. Patent Law

1. European Patent Convention

66 **Swiss law aligns** so strongly with the **EPC/EPO rules** on software patenting[106] that this section can focus on the latter. Additional details on the EPO practice and the EU legislation discourse are provided in the annex (see annex, I.1, I.2).

67 According to the two steps of its **COMVIK approach** for computer-implemented inventions,[107] the EPO first assesses whether the claimed invention is of a technical character and whether an issued patent would therefore not protect software "as such". The second step of its assessment looks at the patent requirements of novelty and inventive step.

a) Protected Subject Matter

68 Art. 52 EPC defines the patent granting criteria which must be met by the subject matter for which protection is sought. The concept of "**invention**" is further clarified in para. 2 of Art. 52 EPC, where a negative definition is given of types of subject matter that are not considered to be inventions, such as mathematical methods (Art. 52(2)(a) EPC) and **programs for computers (Art. 52(2)(b) EPC)**. While the Swiss Patent Act, unlike the EPC and TRIPS,[108] does not explicitly refer to a "field of technology" to which an invention must belong, it considers its **technical character** a prerequisite for patentability.[109] As per a classical definition by the German Federal Court of Justice, with which the EPO[110] and Swiss jurisprudence[111] concur, a technical invention is "a teaching to methodically utilize controllable natural forces to achieve a causal, perceivable result".[112]

69 Even though it may fall under the above definition,[113] Art. 52(2)(c) EPC declares "programs for computers **as such" not to be patentable** for lack of technicality.[114] Software can be patentable as part of a **"computer-implemented invention" (CII)**, viz. an invention "which involves the use of a computer, computer network or other programmable apparatus, where one or

---

105 HEINRICH, in: PatG/EPÜ Kommentar, Art. 1 PatG/Arts. 52, 56, 57 EPÜ paras. 37, 53.

106 HEINRICH, in: PatG/EPÜ Kommentar, Art. 1 PatG/Arts. 52, 56, 57 EPÜ paras. 37, 53. For the concurrent intention of the Swiss legislature, see BBl 1976 II 1, 57, 67.

107 TBA, 26 September 2002, T 641/00 – *Two identities/COMVIK*; SCHWARZ, GRUR 2013, 1419.

108 Art. 52(1) EPC, Art. 27(1) TRIPS.

109 HEINRICH, in: PatG/EPÜ Kommentar, Art. 1 PatG/Arts. 52, 56, 57 EPÜ para. 10; MARBACH et al., 7; BGer sic! 1997, 77 et seq. para. 4.

110 EBA, 9 December 2010, G 2/07 – *Broccoli/PLANT BIOSCIENCE*, para. 6.4.2.1.

111 BGer sic! 1997, 77 et seq. para. 4.

112 Federal Court of Justice ("Bundesgerichtshof, BGH"), X ZB 15/67, 27 March 1969 – *Rote Taube*, para 3; translated to English in: IIC 1970, Vol. 1, No. 1, 136 – *Red Dove*, para. 3.

113 HEINRICH, in: PatG/EPÜ Kommentar, Art. 1 PatG/Arts. 52, 56, 57 EPÜ paras. 37, 40.

114 HEINRICH, in: PatG/EPÜ Kommentar, Art. 1 PatG/Arts. 52, 56, 57 EPÜ paras. 37, 53; STEINBRENNER, in: Singer et al., Art. 52 EPÜ para. 37; BGE 98 I b 396 para. 4; KÖPFLI/CARREIRA in: Hilti et al., 155.

more features are realised wholly or partly by means of a computer program".[115] For the CII, it is critical that the software, when run on a computer, achieves a technical effect which goes beyond the normal physical interactions between the program and the computer ("further technical effect").[116]

70 The **technical character of a computer-implemented invention can,** according to IPI guidelines, **stem** from any of the following four aspects (no cumulation required): (i) the task underlying the claimed computer-implemented invention and solved by it; (ii) the invention's means, i.e. the technical features which constitute the solution to the underlying problem; (iii) the effects achieved by the solution to the problem; (iv) the need for technical considerations in order to arrive at the claimed computer-implemented invention.[117] The initial threshold for software is decidedly low: a computer program, even if it is claimed as a method, has a technical feature in so far as it runs on a computer (hardware).[118]

71 The **patent eligibility step assesses the technical character** of the CII as a whole and without having regard to prior art.[119] Even where non-technical features (i.e. features which taken in isolation are not patentable under Art. 52(2) EPC[120]) form a large part of the claimed subject matter,[121] their **combination with certain technical features** saves the invention's patent eligibility.[122] This is one of the reasons why claims to computer-implemented inventions are typically "mixed invention" claims which relate to a combination of non-technical features (e.g. a computer program) and technical features (e.g. computer, server, mobile phone and special purpose hardware).[123] Nor do the technical features need to be novel to secure patent eligibility/technical character. The novelty assessment instead forms, part of the second COMVIK step.

72 The patentability exclusion for computer programs **interacts with the exclusions for mathematical methods and business methods (Art. 52(2)(a), (c) EPC)**. Algorithms can be understood as mathematical methods that can be combined in a more or less complex way to form a computer program.[124] As such, mathematical calculations, the carrying out of algorithmic instructions, methods of performing mental acts, or the elaboration and deployment of business methods can lack

---

115 EPO Guidelines, Index for Computer-Implemented Inventions.

116 EBA, 12 May 20210, G 3/08 – *Computerprogramme,* paras. 10.2.1, 10.8.4; see also EPO Guidelines G-II, 3.6.

117 IGE Richtlinen, 2.1.1; see also KÖPFLI/CARREIRA in: Hilti et al., 155.

118 TBA, 21 April 2004, T 258/03 – *Auction method/HITACHI*.

119 The first hurdle is taken without reference to the prior art; see EBA, 10 March 2021, G 1/19 – *Pedestrian simulation*, para. 78; TBA, 26 September 2002, T 641/00 – *Two identities/COMVIK*, para. 6; EPO Guidelines, G-II, 2; MELULLIS, in: EPÜ Kommentar Benkard, Art. 52 EPÜ paras. 61, 63 et seq.; VALLONE, sic! 2021, 510.

120 EPO Guidelines, G-II, 2.

121 On the relevance of such settings, see EPO Guidelines, G-VII, 5.4; HEINRICH, in: PatG/EPÜ Kommentar, Art. 1 PatG/Arts. 52, 56, 57 EPÜ para. 14 and 46; KÖPFLI/CARREIRA in: Hilti et al., 155; TBA, 21 April 2004, T 258/03 – *Auction method/HITACHI*, para. 3.5.

122 See TBA, 15 April 1993, 1, T 110/90 – *Editable Document Form/IBM*, para. 5; HEINRICH, in: PatG/EPÜ Kommentar, Art. 1 PatG/Arts. 52, 56, 57 EPÜ para. 46; MELULLIS/KOCH, in: EPÜ Kommentar Benkard, Art. 52 EPÜ para. 277.

123 MINSSEN/ABOY, JIPLP 2021, 633. This has also earned COMVIK the nickname "any hardware approach", EBA, 10 March 2021, G 1/19 – *Pedestrian simulation*, paras. 28, 29; affirmed in TBA, 7 November 2022, T 702/20 – *Sparsely connected neural network/MITSUBISHI*, para. 11.1.

124 SCHUMACHER, 117.

technicality and, consequently, patentability.[125] Today's frequent use of software and computer hardware for such activities does not, in and of itself, alter this result.[126]

73 AI and machine learning are based on computational and classification models/algorithms.[127] The **EPO classifies AI technologies** not as computer programs but as mathematical methods in the form of computational models which are not patentable as such.[128] However, if an AI/machine learning process uses technical means (e.g. a computer), it is of a technical character, which means it is not excluded from patentability under Art. 52(2) and (3) EPC.[129] In this respect, AI technologies are no different from mathematical methods and (conventional) computer programs.[130] According to the EPA, the processing and provision of the training data, as well as the training of the system, can also contribute to the technical character of an AI system, if they support achieving that technical purpose.[131]

b) Requirements for Protection

aa) Novelty and Inventive Step

74 Beyond its technical character, a patentable invention must be new and must embody an inventive step. Regarding CII, the **second COMVIK step looks at all technical features of the claimed invention and considers whether these requirements are met**. The same goes for features that contribute, in the overall context of the invention, to its technical nature, even though they may lack technicality when examined in isolation.[132] Features that do not contribute to the invention's technical character are excluded from the novelty and inventiveness assessment.[133] Regarding the relevant set of features, prior art matters for whether this set qualifies as novel and inventive (i.e. prior art does count for the second COMVIK step).

bb) Industrial Applicability

75 The invention must be industrially applicable (Art. 57 EPC), i.e. intended for manufacture or use in industry, executable and repeatable as often as desired.[134] The results of the technical

---

125 For details, see MELULLIS, in: EPÜ Kommentar Benkard, Art. 52 EPÜ paras. 234 et seq.

126 As case law examples, see TBA, 5 October 1988, T 22/85 – *Document abstracting and retrieving*; Federal Patent Court ("Bundespatentgericht, BPatG"), 30 July 2002 – 17 W (pat) 66/01 (*Fuzzy Clustering*); HEINRICH, in: PatG/EPÜ Kommentar, Art. 1 PatG/Arts. 52, 56, 57 EPÜ paras. 14 and 28; MELULLIS, in: EPÜ Kommentar Benkard, Art. 52 EPÜ paras. 235, 237, 240.

127 EPO Guidelines, G-II, 3.3.1.

128 EPO Guidelines, G-II, 3.3.1; SCHUMACHER, 123.

129 EPO Guidelines, G-II, 3.3.

130 SCHUMACHER, 125.

131 EPO Guidelines, G-II, 3.3.1.

132 TBA, 21 April 2004, T 258/03 – *Auction method/HITACHI*; EBA, 10 March 2021, G 1/19 – *Pedestrian simulation*, para. 31; EPO Guidelines, G-II, 2 and G-VII, 5.4; TBA, 26 September 2002, T 641/00 – *Two identities/COMVIK*, paras. 6 et seq.; MELULLIS, in: EPÜ Kommentar Benkard, Art. 52 EPÜ para. 285; BALDUS, GRUR Int. 2021, 960.

133 EPO Guidelines, G-II, 2 and G-VII, 5.4; TBA, 26 September 2002, T 641/00 – *Two identities/COMVIK*, para. 6.

134 TROLLER, 47.

teaching must occur regularly and not just with a greater or lesser degree of probability.[135] This **requirement will rarely cause difficulties with software**.

cc) Sufficiency of Disclosure

76 Pursuant to Art. 83 EPC, the invention must be disclosed clearly and completely so that it **can be carried out by a person skilled in the art**. In doing so, not only must the structure be described, but also the functionality.[136] The EPO expressly points out that a clear, **detailed functional description of software can be much more expedient than an "overly precise" structural description**.[137] Disclosure of the program structure and, in the case of "learning software", the training data may be sufficient, provided that this enables replication.[138]

77 Regarding AI systems, it is often difficult to understand or illustrate exactly how they work, or even what their detailed structure/architecture is **("black box AI").** Even in the case of "white box AI", structures/architectures can be very complex to illustrate.[139] Given these difficulties, and to avoid a general decline of protectability standards,[140] additional parameters are being considered. In particular, **source code disclosure may be necessary**.[141] Furthermore, parameters such as **weight settings in a neural network or the optimisation algorithm** may be disclosable in code language or other tangible forms.[142] The Board of Appeal of the European Patent Office even had recourse to the disclosure of **training data** of an AI system with neural networks.[143]

c) Application to Simulations and AI Systems

78 Before AI/neural networks became a prominent topic, the **patentability of simulations** was arguably the **most intensely discussed issue** in the field of computer-implemented inventions.[144] Simulations are used to recreate a system with its dynamic processes in an experimental model in order to gain knowledge that can be transferred to reality.[145] Since the simulation is computer based, the EPO's examination practice developed for computer-implemented inventions in patent law applies, including the **two-hurdle approach**.[146] As a core issue, simulations consist of a mathematical model based on the reality to be simulated. Both this mathematical method and the computer program "calculating" the simulation constitute elements lacking technical character, and are therefore excluded as such from patentability (Art. 52(2) EPC). As the EPO has put it, a simulation can be

---

135 BGE 120 II 312, para. 2.

136 EPO Guidelines, F-III, 1.

137 EPO Guidelines, F-III, 1.

138 HEINZE/ENGEL, in: KI-Rechtshandbuch, § 10 para. 66; MÉNIÈRE/PIHLAJAMAA, GRUR 2019, 332, 335; SCHUMACHER, 133.

139 MÉNIÈRE/PIHLAJAMAA, GRUR 2019, 332, 335; SCHUMACHER, 133.

140 HEINZE/ENGEL, in: KI-Rechtshandbuch § 10 para. 66; SCHUMACHER, 134.

141 SCHUMACHER, 134.

142 HEINZE/ENGEL, in: KI-Rechtshandbuch § 10 para. 66; MÉNIÈRE/PIHLAJAMAA, GRUR 2019, 332, 335; SCHUMACHER, 135.

143 TBA, 12 May 2020, T 0161/18 – *Äquivalenter Aortendruck/ARC SEIBERSDORF*, para. 2.2.

144 See for example BALDUS, GRUR Int. 2021; BENNETT, sic! 2020; GROB, GRUR-Prax. 2021; MINSSEN/ABOY, JIPLP2021; VALLONE, sic! 2019.

145 VALLONE, sic! 2019, 660.

146 EBA, 10 March 2021, G 1/19 – *Pedestrian simulation*, para. 136; VALLONE, sic! 2019, 664; BENNETT, sic! 2020, 245.

considered a virtual invention[147] that for patentability requires a technical contribution beyond the virtual space.[148]

79 Such a **technical feature**, which is necessary to escape Art. 52(2) EPC but also to secure novelty and non-obviousness (for which only technical features can be considered), **can lie** in the simulation's purpose to prepare, improve or test the manufacture of a specific product.[149] In the EPO's words, the simulation then pursues an "adequately defined technical purpose for a computer-implemented method, provided that the method is functionally limited to that technical purpose".[150]

80 Where the simulation lacks such a clear tie to the manufacture of a specific product, it can **possess technicality due to a direct link with physical reality**.[151] Furthermore, a sufficient **"potential"** technical effect[152] can be present when (i) "the technical effect that would result from the intended use of the data could be considered "implied by the claim"; or (ii) "the intended use of the data (i.e. the use in connection with a technical device) could be considered to extend across substantially the whole scope of the claimed data processing method."[153] Finally, the EPO exceptionally accepts "**virtual**" or "implied" technical effects, such as the (measurement-based) determination of the physical state of an object (e.g. its temperature).[154]

81 **AI is assessed, in principle, as a mere mathematical method** which nonetheless may contribute, , to the technical character of an invention through "application to a field of technology" or through "adaptation to a specific technical implementation", especially if combined with other functions.[155] Neural networks define a class of mathematical functions which are not patentable as such.[156] Like other "non-technical" matter, they can only be taken into account in assessing the inventive step if they are **used to solve a technical problem** (e.g. because they were trained with specific data for a particular technical task[157]). Where a patent claims a neural network apparatus implemented on a computer, it can pass the COMVIK approach's first hurdle under the "any hardware" test.[158] However, it does not suffice that "the claim as a whole specifies abstract computer-implemented mathematical operations on unspecified data, namely that of defining a class of approximating functions (the network with its structure), solving a (complex) system of (non-linear) equations to obtain the parameters of the functions (the learning of the weights), and using it to compute outputs for new inputs".[159] The application must also claim solution of a specific technical problem,

---

147 See TBA, 13 December 2006, T 1227/05 – *Circuit simulation I/Infineon Technologies*, para. 3.2.2; VALLONE, sic! 2019, 664.

148 See TBA, 13 December 2006, T 1227/05 – *Circuit simulation I/Infineon Technologies*, para. 3.3; VALLONE, sic! 2019, 664.

149 TBA, 13 December 2006, T 1227/05 – *Circuit simulation I/Infineon Technologies*, para. 3.1; TBA, 19 January 2017, T 625/11 – *Areva*, para. 8.1.2.; see VALLONE, sic! 2019, 665.

150 TBA, 13 December 2006, T 1227/05 – *Circuit simulation I/Infineon Technologies*, para. 3.1.

151 EBA, 10 March 2021, G 1/19 – *Pedestrian simulation*, para. 88.

152 EBA, 10 March 2021, G 1/19 – *Pedestrian simulation*, paras. 89 et seqq.

153 EBA, 10 March 2021, G 1/19 – *Pedestrian simulation*, para. 94.

154 EBA, 10 March 2021, G 1/19 – *Pedestrian simulation*, paras. 97 et seqq.

155 TBA, 9 May 2018, T 2330/13 – *Checking selection conditions/SAP*; EPO Guidelines, G-II, 3.3.

156 TBA, 7 November 2022, T 702/20 – *Sparsely connected neural network/MITSUBISHI*, para. 19.

157 TBA, 7 November 2022, T 702/20 – *Sparsely connected neural network/MITSUBISHI*, paras. 11 et seq.

158 TBA, 7 November 2022, T 702/20 – *Sparsely connected neural network/MITSUBISHI*, paras. 10, 13.

159 TBA, 7 November 2022, T 702/20 – *Sparsely connected neural network/MITSUBISHI*, para. 19.

such as the overfitting of the neural network.[160] Sufficient specification can be derived from describing the training data and the technical task addressed.[161]

d) Acquisition of Rights, Ownership, Transferability

82 Swiss law, which is the norm at least for Continental patent laws, entitles the **human inventor** to a patent on that invention.[162] Joint inventorship leads to **joint patent entitlement**.[163] One of the joint owners may exercise the patent rights only with the consent of the others; however, each owner may independently dispose of his/her part, or bring an action for patent infringement.[164] These general rules also apply to software patents.[165]

83 Equally, software-related patents share the character of other patents in that the rights emanating from them cover, first and foremost, the **commercial use of the patented invention**.[166] Moral rights do not loom large in patent law, with their main manifestation being an inventor's right to be named as such in the patent application and register entry.[167] Rights to the grant of a patent and issued patents are, together with the use/commercial rights embodied in them, **fully assignable**.[168] The inventor can waive even his/her right to be named, albeit only ex post.[169]

84 Contrary to Swiss copyright law, Swiss patent law does not contain a specific provision on software developed by employees. Such software is, however, subject to more general provisions on **employee inventions** in Art. 332 CO, the high practical relevance of which results from the fact that most software development today is performed in a corporate setting. According to Art. 332 CO,[170] inventions made by an employee in the course of his/her work for the employer and in performance of his/her contractual obligations belong to the employer. There is no obligation for the employer to specifically remunerate such inventions. Once the employee acquires a patent on the invention, it is assigned to the employer by operation of law. By written agreement, an employer may reserve the right to acquire, through notice to the employee and for adequate payment, inventions produced by an employee in the course of his/her work for the employer but not in performance of his/her contractual obligations. No such reservation mechanism exists for inventions made outside an employee's course of work, but some contend the employee needs to at least inform the employer about such inventions, provided they are of relevance for the employer's business.

---

160 TBA, 7 November 2022, T 702/20 – *Sparsely connected neural network/MITSUBISHI*, para. 19.

161 TBA, 7 November 2022, T 702/20 – *Sparsely connected neural network/MITSUBISHI*, para. 20.

162 BREMI, in: SHK, Art. 3 PatG paras. 14, 25; MELULLIS/KOCH, in: EPÜ Kommentar Benkard, Art. 60 EPÜ para. 4.

163 BREMI, in: SHK, Art. 3 PatG paras. 50 et seq., 53; MELULLIS/KOCH, in: EPÜ Kommentar Benkard, Art. 60 EPÜ para. 16.

164 Art. 3(2), 33(2) PatA; on the legal relationship between joint owners, see BREMI, in: SHK, Art. 33 PatG paras. 11 seq. On the similar legal situation under the EPC, see SINGER/STAUDER, in: Singer et al., Art. 60 EPÜ paras. 7 et seqq.

165 STRAUB, Softwareschutz, 513 et seqq.

166 See Art. 8(1) PatA.

167 Art. 5 PatA, Art. 62 EPC.

168 SCHWEIZER, in: SHK, Art. 33 PatG paras. 9 et seqq.

169 Art. 6 PatA; Rule 20 Implementing Regulations to the EPC; BREMI, in: SHK, Art. 6 PatG para. 5.

170 For details on the following – from a software perspective – see STRAUB, Softwareschutz, paras. 514 seqq.; BREMI, in: SHK, Art. 3 PatG paras. 26, 41 et seq.

85 With regard to whether patent applications may designate AI systems as inventors, Swiss courts have not yet decided. It seems likely that they would follow the lead taken by the EPO and certain EU Member State courts in the DABUS line of cases.[171] Briefly, this approach does **not accept registration of an AI system as the inventor** and rejects as inadmissible patent applications to this effect.[172] However, at least the EPO and the German Federal Patent Court arguably indicate readiness to accept patent applications which describe, in their specifications, the inventive role of AI, while designating a human as the (nominal) inventor.[173]

e) Effects of Protection

aa) Economic and Moral Rights

86 Economic aspects form the core of the exclusive right granted to inventors, namely the right to **exploit their patented inventions commercially**. The right to commercial use of a software-related patent encompasses the manufacturing, storage, offering, commercialization, importing, exporting and transport of patent-implementing products, as well as possession for any of these purposes.[174] A form of use essential to software is its multiplication through copying.[175]

87 Where the patent covers a manufacturing process, the effects of the patent extend not only to the performance of that process but also to the products directly obtained by it.[176] **Data generated by the running of patent-protected software, however, do not qualify as process products** in that sense.[177]

88 After the **exhaustion** of patent rights with regard to a particular product, further use of this product does not infringe the respective patent. In principle, this includes repairs made to the product. As regards software, bug patches for software are – summarily speaking - considered repairs, whereas modifying updates can fall outside the exhaustion safe harbour.[178]

89 While the EPC does not explicitly address moral rights, it indirectly recognizes the importance of inventorship and the inventors' contributions to the patented invention. As such, inventors are typically considered to have **very limited moral rights** associated with their inventions, which include the right to be identified as the inventor.[179]

---

171 ABBOTT, B.C. L. Rev. 2016, 1079; PICHT et al., 10 et seqq.; STIERLE, GRUR Int. 2020, 918; ENGEL, GRUR Int. 2020, 1123; BONADIO et al., Int. Prop. Q. 2021, 56 et seqq.; KIM et al., 2 et seqq.; *Thaler v. Vidal,* Brief of Amici Curiae.

172 EPO, Grounds for decisions of 27 January 2020 on EP18275163.6, paras. 34 et seqq. and EP18275174.3, paras. 35 et seqq.; BPatG, 11 November 2021 – 11 W (pat) 5/21 (*Fractal Container)*, para II.2.a; BPatG, 21 June 2023 – 18 W (pat) 28/20 (*Neural Flame*), para. II.2.

173 EPO, Grounds for decisions of 27 January 2020 on EP18275163.6, para 37 seq. and EP18275174.3, 38 seq.; BPatG, 11 November 2021 – 11 W (pat) 5/21 (*Fractal Container)*, paras. II.1 and II.2.a.bb; BPatG, 21 June 2023 – 18 W (pat) 28/20 (*Neural Flame*), paras. II.1 and II.2.

174 Art. 8 PatA; Art. 64(1) EPC.

175 STRAUB, Softwareschutz, para. 529.

176 Art. 8a PatA; Art. 64(2) EPC.

177 CALAME, 449.

178 STRAUB, Softwareschutz, para. 528 w.f.r.

179 SHEMTOV, inventorship, 22 et seq; FROMER, Va. L. Rev. 2012, 1771 et seqq.

bb) Limitations

90 Various limitations balance software-related patent protection against legitimate interests in using the protected subject matter.[180]

aaa) Private Use/Copy

91 Art. 9 para. 1 lit. a PatA creates a safe harbour for private use, whereby the effects of the patent do not extend to **non-commercial use** in accordance with Art. 8 PatA.[181] Private use and private copying of patented software are therefore permitted.[182]

bbb) Scientific Research

92 The patent may be used for research and testing (Art. 9(1) lit. b PatA), even for commercial purposes.[183] For such purposes, methods of **reverse engineering** may be applied to computer programs protected by patent law.[184] Using various reverse engineering techniques – in particular, decompilation – it is also possible to reconstruct the original source code.[185] However, reverse engineering **must also consider copyright law boundaries** (see below, D.IV.1.d)bb)ccc), D.IV.1.d)bb)fff)).[186]

ccc) Continued Use

93 The patent cannot be held against a person who has already commercially implemented the invention, in Switzerland and in good faith, before the filing or priority date (Art. 35 PatA). The right of use is generally **limited to the invention actually pre-used[187] and allows use for business purposes** (Art. 35(2) PatA). Doctrinally speaking, this provision establishes a royalty-free statutory licence.[188]

ddd) Dependent Inventions

94 If a patented invention **cannot be used without infringing an earlier patent**, the owner of the later patent is entitled to a non-exclusive licence to the extent necessary for its use, provided that the **subsequent invention constitutes a notable technical advance of considerable economic importance** in comparison with that of the earlier patent (Art. 36(1) PatA).[189] This ensures that existing patents do not excessively hinder significant technical progress in the interests of innovation, particularly for economic reasons.[190]

---

180 For a more detailed overview on the following, see STRAUB, Softwareschutz, paras. 530 et seqq. w.f.r.

181 HESS-BLUMER, in: SHK, Art. 9 PatG para. 8.

182 STRAUB, Softwareschutz, para. 531.

183 HESS-BLUMER, in: SHK, Art. 9 PatG para. 29; HEINRICH, in: PatG/EPÜ Kommentar, Art. 9 PatG para. 3.

184 STRAUB, Softwareschutz, para. 533.

185 STRAUB, Softwareschutz, para. 272.

186 STRAUB, Softwareschutz, paras. 271 et seqq.

187 STRAUB, Softwareschutz, para. 532; GASSER, in: SHK, Art. 35 PatG para. 37.

188 GASSER, in: SHK, Art. 35 PatG para. 2.

189 With regard to software, see BERGER, 58; STRAUB, Softwareschutz, para. 534.

190 GASSER, in: SHK, Art. 36 PatG para. 3.

eee) Lack of Execution of the Invention

95 If the patent proprietor does not use the patent, including licensing, and if there are **no objective grounds for inertia**, an action for the grant of a **non-exclusive licence** may be brought pursuant to Art. 37 PatA.[191] In practice, this mechanism is **of little relevance**.[192]

fff) Public Interests

96 If public interests outweigh those of the patent holder, **compulsory licences** may be granted.[193] To our knowledge, this provision has **never been applied** in practice in the field of IT.[194]

cc) Term of Protection

97 For Swiss and European patents, the term of protection is **20 years** (Art. 14 PatA, Art. 63(1) EPC) from the date of application (Art. 56 PatA, Art. 80 EPC). This corresponds to the international standard, defined in Art. 33 TRIPS Agreement as a minimum term.[195]

2. Comparative Findings

a) Protected Subject Matter

98 **Software as such is not patentable** in any of the assessed jurisdictions, and all of them have different approaches as to when the requirements for patentability are met. In the EPO Jurisdictions, the UK and China, software is only patentable if there is a "technical character". Under the US system, there must be an inventive concept that leads to "significantly more" than the abstract idea itself. In the other jurisdictions considered, software and hardware components must "cooperate" (Japan) or "interact" (Singapore) when solving the technical problem at hand.

99 The overarching goal of patenting standards is to **avoid overly broad patents and to prevent downstream inventors from being hindered** in their activities. Additionally, there is an effort to steer clear of creating dense patent thickets. However, fluctuating standards among courts and patent offices led to **significant uncertainty regarding what is patentable and what is not**. This issue becomes particularly problematic in the realm simulations.

100 Court case law is visibly driven by the fact that **software simply does not fit neatly in the patent systems**. This is evidenced by the lengthy sections in patenting guidelines across all jurisdictions, which in many respects still fall short of legal certainty.

b) Requirements for Protection

101 **No** analysed **jurisdiction** outside the EPC realm (UK, US, Japan, China and Singapore) **differs significantly from the EPC** in terms of protection requirements. Requirements of "usefulness"/"utility" in the US or "practical use" in China are recognized as those corresponding to industrial applicability under the EPC, UK, Japan and Singapore.

---

191 STRAUB, Softwareschutz, para. 535.

192 SEITZ, in: SHK, Art. 37 PatG para. 5; HEINRICH, in: PatG/EPÜ Kommentar, Art. 37 PatG para. 3.

193 Regarding software, see BERGER, 58; STRAUB, Softwareschutz, para. 536.

194 STRAUB, Softwareschutz, para. 536.

195 BERGER, 58; STRAUB, Softwareschutz, para. 544.

c) Acquisition of Rights, Ownership, Transferability

102 In all examined jurisdictions, a **human inventor is required** and must be mentioned in the patent application. **AI systems or entities cannot be designated as inventors**. The threshold for inventorship differs only semantically. Generally, an inventor is defined as someone who contributes in a substantive manner to the part of the invention which distinguishes it from the prior art, rendering it "inventive". The USPTO is at present the only patent office that provides guidelines specifically addressing the question of where the significant contribution may lie when an invention is made with the assistance of an AI system. **Simply identifying the problem and feeding it to an AI system is not enough. However, where an individual contributes substantially to the crafting of a prompt, he/she may be considered the inventor.**[196] Simply maintaining **"intellectual dominance" over an AI system does not make a person an inventor** of the inventions made by the AI system. By simply **owning or overseeing such a system one does not reach the threshold** of a significant contribution.[197]

103 The essential objective in mentioning the inventor in the application is to grant him/her the initial right to obtain the patent and the moral rights that come with it. **In practice, exclusive rights will often be assigned to the applying entity**. Mentioning the inventor serves the main purpose of creating a nexus between the inventive individual and the registering entity owning the patent.[198]

d) Effects of Protection

104 In all the examined jurisdictions, economic rights allow rights holders the exclusive right to commercially exploit the patented invention, i.e. the **right to prohibit others from making, using, offering for sale, selling or importing the patented invention** without authorisation.

105 In all jurisdictions, **private use** of the patented invention for non-commercial purposes does not infringe the patent. Likewise, all the jurisdictions examined have a limitation for **prior use,** scientific research and experimental use. The limitations for **scientific research and experimental use** vary from country to country in terms of their scope. The US has a narrower interpretation than Europe or Japan. In the US, commercial use is not covered by the exception, which means that even university research is an infringing act.[199] Japan, on the other hand, does not distinguish based on the type of organisation conducting the experiment or research. In China, the patent will only be infringed if the use of a patented technology was not for the improvement of that technology and the result was not related to it.[200]

106 In addition, the UK, the US, Japan, China and Singapore have established **compulsory licences to prevent anti-competitive** behaviour and, for the most part, for instances of state emergencies or similar cases.

---

196 USPTO, Inventorship Guidance for AI-assisted inventions, 10048 (criterion 2), with reference to: *Shatterproof Glass Corp. v. Libbey-Owens Ford Co.*, 758 F.2d 613 (Fed. Cir. 1985), 624 ("An inventor 'may use the services, ideas, and aid of others in the process of perfecting [their] invention without losing [their] right to a patent."), (quoting *Hobbs v. U.S. Atomic Energy Comm.*, 451 F.2d 849, 864 (5th Cir. 1971)).

197 USPTO, Inventorship Guidance for AI-assisted inventions, 10048 (criterion 5), with reference to: *Verhoef*, 888 F.3d 1362, 1367 (court refused to endorse the "intellectual domination" language and emphasized that the person who conceives of the invention is the inventor).

198 See LIM, Akron L. Rev. 2019, 857 et seq.

199 *Madey v. Duke University*, 307 F.3d 1351 (Fed. Cir. 2002), 1360 et seq.

200 BAILEY/WANG, in: Luginbühl/Ganea, 134.

107 Certain countries, such as the US and Japan, have not adopted express legislative provisions on **exhaustion**, leaving this matter to case law. The same goes for European patents under the EPC,[201] while Art. 6 Unitary Patent Regulation (1257/2012) contains an explicit exhaustion provision for unitary patents.

108 All examined jurisdictions adhere to a minimum term of protection of **20 years** from the year the patent application was filed as set out in Art. 33 TRIPS.

IV. Copyright Law

1. Swiss Law

a) Protected Subject Matter

109 The Swiss Copyright Act (CopA) grants protection for **literary and artistic works** (Art. 1 lit. a CopA). Such works are intellectual creations with individual character, irrespective of their value or purpose (Art. 2(1) CopA). If these conditions are met, the literary or artistic work is copyright protected.

110 According to Art. 2(3) CopA, **computer programs are also deemed to be works**. This "legal fiction" allowed the legislator to include the protection of computer programs in the framework of the existing copyright law, despite fundamental differences between computer programs and literary and artistic works.

111 The Swiss legislator deliberately **refrained from including a legal definition of the term "computer program"** in the CopA in order to cover future technical developments.[202] This approach has proven to be helpful, as it makes it possible to capture AI systems and to grant copyright protection for this specific type of software. Copyright scholars do not deviate from the generally accepted definition of computer programs (see above, B.I).[203]

112 As opposed to patent law which captures technical features of computer programs, the subject matter of copyright is the **implementation of a computer program in the source code and its representation in the (binary) object code**.[204] By focusing on the code layer, copyright law applies a "**linguistic approach**", thereby facilitating the integration of computer programs into copyright law by allowing to think of them in terms of linguistic works.

113 Copyright law does not protect mere ideas, concepts and instructions; it only captures the **expression** of the human mind **in a specific form, not the underlying idea or concept itself**.[205] Applying this fundamental principle of copyright law to software means that abstract methods and ideas, in particular **underlying algorithms, are not (directly) protected**.[206] From the perspective of copyright law, the creative achievement and protected subject matter is the **expression of the**

---

201 HENKE, in: EPÜ Kommentar Benkard, Art. 64 EPÜ paras. 61, 637 et seq.

202 BBl 1989 III 477, 522.

203 RAUBER, 127 et seq.; EGLOFF, in: Barrelet/Egloff, Art. 2 URG para. 32.

204 HILTY, sic! 2013, 706; EGLOFF, in: Barrelet/Egloff, Art. 2 URG para. 32; For details about source and object code see above paras. 27 et seqq.

205 BBl 1989 III 477, 521; REHBINDER et al., in: OFK, Art. 2 URG para. 2; EGLOFF, in: Barrelet/Egloff, Art. 2 URG para. 3; STRAUB, Informatikrecht, 11; LUTZ, GRUR Int. 1993, 653 et seq.; see also Art. 1(2) Dir. 2009/24.

206 BBl 1989 III 477, 523; LUTZ, GRUR Int. 1993, 653 et seq.

algorithm in a specific form, i.e. the written form of a series of instructions in a specific programming language. However, since the scope of protection of copyright law is not limited to identical copies, it also provides protection at a more abstract level, such as the detailed story line of a book (e.g. a book with a story line identical to the first book in the Harry Potter series would infringe the copyright in the latter book even if the former book expressed the story line in different words). This equally applies to computer programs. Even if the underlying algorithm is not directly protected by copyright, it can be indirectly protected against a third-party computer program using identical steps to solve a problem, albeit with slight variations on the code expression level.

114 As with other works, the scope of protection of computer programs is not limited to identical copies but comprises **sufficiently similar programs**.[207] By expanding the protection to similar programs, the **underlying algorithm is protected indirectly**, since a third party using similar code in the same programming language or simply translating the code into another programming language will use the algorithm underlying the original code. As a result, and despite copyright's linguistic approach, the scope of protection of a computer program may capture the underlying logic to a certain extent.[208] What matters is that the algorithm as such is not protected by copyright, independent of its representation in source code.

b) Requirements for Protection

aa) Intellectual Creation

115 The idea of **human authorship** is one the fundamental principles of the Continental European *droit d'auteur* systems, including Swiss copyright law.[209] One of the key rationales for granting copyright protection to literary and artistic works is that human beings have a natural right to control the product of their labour, including intellectual creations (see above, C.II). This rationale is rightfully contested. Copyright law, and more specifically the specific moral rights granted, should allow authors to protect their personality that can be violated by certain uses of their work, e.g. if the work is distorted. The key role of human authorship is mirrored in a series of provisions of the CopA, not least in the **requirement of intellectual creation**.

116 The protection requirement of an intellectual creation ensures that only works created by humans can be works within the meaning of copyright law. The term "intellectual" refers to the activity of one (or more) human being(s) and expresses that the work is based on human will and must be the expression of a human thought.[210] While the CopA explicitly states the requirement of an "intellectual creation" for literary and artistic works, it is silent in this respect for computer programs (Art. 2(3) CopA) and photographs lacking individual character (Art. 2(3bis) CopA). However, an interpretation taking into account the legislator's intent[211] and the structure of Art. 2 CopA leads to the conclusion that the requirement of intellectual creation **also applies to computer programs** and photographs lacking individual character. This view is shared in the legal literature.[212] If the **human mind does not decide on the result, there is no intellectual creation**. Human creators may

[207] HILTY, Urheberrecht, paras. 373 et seqq.

[208] HILTY, sic! 2013, 706 et seq.

[209] BGE 116 II 351 para. 2.b; BGE 74 II 106 para. 3.

[210] BGE 130 III 714 para. 2.1; BGE 130 III 168 para. 4.5.

[211] BBl 1989 III 477, 522; BBl 2018 591, 620.

[212] EGLOFF, in: Barrelet/Egloff, Art. 2(3bis) URG para. 35; STRAUB, Softwareschutz, para. 74; RAUBER, Computersoftware, 130.

use all sorts of tools (e.g. computer programs, including AI systems) to create a work, but the work must still be the product of human will. To meet this condition, a sufficient human participation in the creation of the work is required. Consequently, **works autonomously created by machines are not protected under the CopA.**[213]

117 Applying this traditional interpretation, it seems to be generally accepted in the Swiss legal literature that **literary and artistic works are not protected by copyright if they have been autonomously generated by an AI system**, due to the lack of a human author.[214] This also applies to computer programs.[215] However, if an AI system is used only as a tool and human input remains substantial, the resulting computer program is protected.[216] As mentioned above, it is foreseeable that AI systems will cross the threshold to autonomous software generation (see above, B.V).

bb) Individual Character

118 The protection requirement of an individual character should draw a line between very simple creations and works that achieve the **level of creativity required for copyright protection**. Given the broad rights and the long protection conferred by copyright, the threshold for meeting this requirement should not be set too low.[217] How this threshold is to be defined in theory and how the existence of **individual character is to be determined in individual cases is disputed**. Numerous approaches have been developed by scholars and applied by courts, but so far none has gained general acceptance.

119 In recent decisions, the Federal Supreme Court applied an approach that is closely aligned with the wording of the protection requirement. According to the Court, the individual character does not require originality in the sense of a personal imprint of the author but allows to **distinguish protected works from banal and mere routine creations**.[218] The individual character results from the variety of decisions made by the author as well as from surprising and unusual combinations, which "make it seem **impossible that the same or essentially the same work would have been created by a third party given the same task**".[219] With this test, the Federal Supreme Court seems to conceive of the individual character as a connection between the author and the work, i.e. the work was created in this way (and not otherwise) precisely because it is the creation of a specific, individual author (or several authors), whereas others would have created a different work if they had been given the same task.

120 The individual character of a work must be considered relative to the respective type of work. While the **abstract measure of the individual character is the same for all works, its concrete assessment must always be carried out in relation to the genre to which the work belongs, taking into account the creative scope and the means that an author could use to create**

---

213 EGLOFF, in: Barrelet/Egloff, Art. 2 URG para. 8; HILTY, Urheberrecht, para. 152; REHBINDER et al., in: OFK, Art. 2 URG para. 2.

214 RAGOT et al., sic! 2019, 574; THOUVENIN/PICHT, sic! 2023, 511; MARMY-BRÄNDLI/OEHRI, sic! 2023, 649.

215 THOUVENIN/PICHT, sic! 2023, 511 et seq.; ROHNER, ZGE 2019, 59; RAGOT et al., sic! 2019, 574.

216 STRAUB, Softwareschutz, paras. 74 et seqq.; SCHÖNBERGER, ZGE 2018, 45; RAGOT et al., sic! 2019, 574.

217 BGE 148 III 305 para. 5.3.

218 BGE 142 III 387 para. 3.1; BGE 136 III 225 para. 4.2.

219 BGE 134 III 166 para. 2.3.2; similar: BGE 142 III 387 para. 3.1; BGE 136 III 225 para. 4.2.

the work.[220] The individual character of a text is therefore not determined according to the same criteria as that of a photograph, film, building – or a computer program.[221]

121 The relative approach of determining the individual character is particularly important for computer programs. In contrast to artistic and literary works, the **authors of computer programs do not strive for creativity in the sense of originality, but rather seek an efficient and often standardised solution** to a problem.[222] Although programmers may implement an algorithm in a given programming language in different ways, programming conventions, standardised programming aids, "best practices" and the like **greatly restrict the programmers' creative freedom** when writing code.[223] Whether code writing is an act of individual character depends on the extent to which programmers are able to make their own decisions when writing code.[224] Drawing parallels between linguistic works and the programming of code helps to further specify whether computer programs possess individual character. In linguistic works, authors constantly make decisions regarding storyline, structure of the text, individual sentences and words used. To some extent, this also applies to computer programs, where programmers decide on the overall structure of the code, lines of code and arrangement of standard sequences.[225] Because of the many choices made by the author(s), longer texts are almost always protected by copyright, as no other author(s) would have made the same choices. This is also true for **computer programs with a reasonably complex structure and a sufficiently large number of lines of code**.[226] As a result, an unnecessarily complex computer program may be protected by copyright while a program with identical or similar functionality, written in the simplest possible way using standard coding practice, may not be. This is hardly convincing and calls for a different criterion to identify code worth protecting (see below, F.II.1.b)).

122 Even before the relative approach was established by the Federal Supreme Court, the Federal Council, scholars and courts used this approach de facto when assessing the individual character of computer programs, stating that a program meets this requirement if it cannot be described as banal or ordinary from the point of view of an expert.[227]

c) Acquisition of Rights, Ownership, Transferability

123 Copyright always originates with the author, i.e., the **natural person who created the work** (Art. 6 CopA). Under Swiss law, copyright cannot originate from legal persons because they cannot be creators; **legal persons therefore can only acquire copyrights derivatively**.[228] These basic principles apply to all types of works, including computer programs. Consequently, the

---

220 BGE 148 III 305 para. 5.3; THOUVENIN, 73; EGLOFF, in: Barrelet/Egloff, Art. 2 URG para. 13.

221 THOUVENIN, 73; EGLOFF, in: Barrelet/Egloff, Art. 2 URG para. 13. As an example for photographs: BGE 130 III 168 para. 4.5.

222 EGLOFF, in: Barrelet/Egloff, Art. 2 URG para. 33; LUTZ, GRUR Int. 1993, 653.

223 HOEREN/WEHKAMP, CR 2018, 3 et seqq.

224 HOEREN/WEHKAMP, CR 2018, 7.

225 OGer ZH, 24 January 2013, LK 100006, para. 5.1; STRAUB, Jusletter, para. 8; See also *Google LLC v. Oracle Am., Inc.*, 141 S. Ct. 1183 (2021), 1205 et seq., where the copied lines of code were considered to establish whether the copying of a fraction of the total lines of code (11,500 out of 2.86 million lines of code) is justifiable under the fair use doctrine. The court concluded that it was.

226 OGer ZH LK 100006 decision of 24 January 2013 para. 5.1; RAUBER, 127.

227 BBl 1989 III 477, 523; NEFF et al., 130 et seqq.; EGLOFF, in: Barrelet/Egloff, Art. 2 URG para. 33.

228 HUG, in: SHK, Art. 6 URG para. 3; EGLOFF, in: Barrelet/Egloff, Art. 6 URG para. 3.

copyright in a computer program developed by a natural person using an AI system always originates in that natural person.

124 Most computer programs are developed by employees. For literary and artistic works, Swiss law does not contain a provision providing for the transfer of the copyright vested in an employee to his/her employer if the employee created a work in the course of work for the employer and in performance of contractual obligations. Unlike for all other types of works, the CopA contains such a provision for computer programs. According to Art. 17 CopA, the **employer alone is entitled to exercise the economic rights to a computer program** if the program was developed under an employment contract in the course of discharging professional duties and in fulfilling contractual obligations. While it is undisputed that Art. 17 CopA does not change the principle that the copyright always originates with its creator (CopA 6),[229] scholars disagree as to whether the copyright in a computer program developed by employees is transferred to their employer in the form of a subrogation[230] or whether an exclusive licence is granted to the employer.[231] In any case, if an employee (or most often several employees) develops a computer program using an AI system, the economic rights of use are vested in the employer.

125 **Economic rights are fully transferable** from the author(s) to any third party.[232] The extent to which **moral rights can be transferred is controversial**. The majority of scholars are of the opinion that such rights are transferable with the exception of those rights that belong to the so-called **core of moral rights**, namely the right to **recognition** of authorship (Art. 9(1) CopA) and the right to oppose any **distortion** of one's work (Art. 11(2) CopA).[233] However, as already explained, moral rights are hardly relevant for the authors of computer programs. As opposed to literary and artistic works, the individual programmers' names are usually not mentioned when the program is published. According to a majority of scholars, a **tacit waiver of the right to be named** can be assumed if **several programmers were involved** in the development of the program.[234] In fact, because moral rights do not appear to play a relevant role with regard to computer programs (see below, E.II), the current regime de facto provides for full transferability of all relevant rights in computer programs.

d) Effects of Protection

aa) Economic and Moral Rights

126 The owner of a copyright in a computer program may assert **all economic rights** under copyright law (Arts. 10(1) et seq. CopA). In addition, and as opposed to the owner of a copyright in literary or artistic works, the owner of a copyright in a computer program also has the **exclusive rental right** (Art. 10(3) CopA). **Several of the economic rights, however, are irrelevant for**

[229] REHBINDER et al., in: OFK, Art. 17 URG para. 1; DE WERRA, in: SHK, Art. 17 URG para. 1; EGLOFF, in: Barrelet/Egloff, Art. 17 URG para. 4.

[230] REHBINDER et al., in: OFK, Art. 17 para. 1; DE WERRA, in: SHK, Art. 17 URG paras. 17 et seqq.

[231] EGLOFF, in: Barrelet/Egloff, Art. 17 URG para. 2.

[232] BGer, 8 May 2008, 4A_104/2008, para. 4.8; REHBINDER et al., in: OFK, Art. 16 URG para. 1; DE WERRA, in: SHK, Art. 16 URG para. 6.

[233] EGLOFF, in: Barrelet/Egloff, Pre. Art. 9–15 URG paras. 4 et seqq. (with a different position, however, in Art. 16 para. 9); REHBINDER et al., in: OFK, Art. 16 URG para. 9; DE WERRA, in: Commentaire romand, Art. 16 URG paras. 21 et seqq.; RIGAMONTI, 266, 305; SEEMANN, 293, 312.

[234] STRAUB, Softwareschutz, para. 136, with further references; NEFF et al., 207 et seq.; RAUBER, Computersoftware, 205; DE WERRA, in: Commentaire romand, Art. 17 URG para. 24.

computer programs, namely the right to recite, perform or present the work (Art. 10(2) lit. c CopA), the right to broadcast and rebroadcast the work, (Art. 10(2) lits. d et seq. CopA), and the right to make the work perceptible (Art. 10(2) lit. f CopA).

127 The owner of a copyright in a computer program may also assert **all moral rights**, namely the right to recognition of ownership (Art. 9(1) CopA), the right of first publication (Art. 9(2) CopA), the right to decide whether, when and under what designation the author's own work is published (Art. 9(2) CopA), the right to decide if the work may be altered (Art. 11(1) lit. a CopA) or used to create a derivative work or be included in a collected work (Art. 11(1) lit. b CopA) and the right to oppose any distortion of the work that is a violation of the author's moral rights (Art. 11(2) CopA). As stated above, however, moral rights are **scarecely relevant** for the authors of computer programs.

128 The principle of **exhaustion**, one of the key principles of copyright law in an analogue world, also applies to computer programs (Art. 12 CopA). However, **making sense** of this principle for **computer programs** that are not sold on a physical storage medium (e.g. CD, DVD or memory stick) is arguably **almost impossible**.[235]

129 As opposed to owners of literary and artistic works, the **owner of a copy of a computer program** also has the explicit, statutory **right to use it** (Art. 12(2) CopA). According to the Swiss Copyright Ordinance (CopO), the right to the intended use of a program includes the loading, displaying, running, transmitting or storing, and production of a copy of the work by the lawful acquirer required in the context of these activities (Art. 17(1)(a) CopO). Moreover, the right to use encompasses observing the functioning of the program, examining or testing it for the purpose of determining the ideas and principles underlying a program element, if this is performed as part of the actions for the intended use (Art. 17(1)(a) CopO). Literary and artistic works are also subject to a right of free consumption **("Werkgenuss")**, according to which the owner of a copyright in such works cannot prohibit the perception of his/her work with human senses.[236] However, this "Werkgenuss" does not take the same form as the right to use a computer program, where said right is **necessary to ensure that the computer program can be operated in accordance with its intended purpose**. This difference again highlights how computer programs are fundamentally different from literary an artistic works.

130 Although neither stated in the Act nor in the Ordinance, many authors are of the opinion that the right of use also contains a somewhat **limited right to bug fixing**.[237] While this interpretation may be considered daring, it caters to an obvious need. As opposed to Swiss law, **European law contains an explicit exception** to the exclusive rights of the owner of a copyright in a computer program for making a **copy** of a computer program and altering the program for the purpose of **error correction** (Art. 5(1) Directive 2009/24/EC). While the lack of an explicit limitation seems not to have caused major practical problems in Switzerland, introducing an explicit provision on bug fixing could provide legal certainty.

---

[235] CJEU, 24 April 2012, *Axel W. Bierbach, administrator of UsedSoft GmbH v. Oracle International Corp.*, C-128/11, paras. 85 et seqq.; HILTY et al., IIC, 263 et seqq.; MALEVANNY, CR 2013, 422 et seqq.; KRÜGER et al., MMR 2013, 760 et seqq.

[236] REHBINDER et al., in: OFK, Art. 10 URG para. 3; EGLOFF, in: Barrelet/Egloff, Art. 10 URG para. 8; HILTY, Urheberrecht, paras. 292 et seqq.; CHERPILLOD, in: Commentaire romand, Art. 10 URG para. 3.

[237] STRAUB, Softwareschutz, para. 222; PFORTMÜLLER, in: SHK, Art. 12 URG para. 18; NEFF et al., 257; CHERPILLOD, SIWR II/1, 285; see also REHBINDER et al., in: OFK, Art. 12 URG para. 9.

bb) Limitations

aaa) Preliminary Remarks

131 The Swiss CopA contains a **number of exceptions and limitations**. Most of them apply to all types of works but have **no relevance to computer programs**, e.g. the right of quotation (Art. 25 CopA; Art. 28(2) CopA), the right of dissemination of broadcast works (Art. 22 CopA), the right to use orphan works (Art. 22*b* CopA), the right to make a picture of works on premises open to the public (i.e. the so-called freedom of panorama [Art. 27 CopA]), or the right to reproduce, present, broadcast or distribute works for the purpose of reporting current events (Art. 28(1) CopA). In addition to this de facto distinction between literary and artistic works and computer programs, some limitations **explicitly apply only to the latter**, namely the right to **decode** a computer program (Art. 21 CopA) and the right to make **backup copies** thereof (24(2) CopA).

bbb) Private Use

132 According to Art. 19(1) CopA, published works may be used for private use. This includes any **personal use** of a work as well as any use within a **circle of persons closely connected** to each other, such as friends and relatives (Art. 19(1) lit. a CopA), any use by teachers and their students for **educational purposes** (Art. 19(1) lit. b CopA), and the copying of a work in **enterprises**, public administrations, institutions, commissions and similar bodies for international information or documentation (Art. 19(1) lit. c CopA).

133 This exception, however, does **not apply to computer programs** (Art. 19(4) CopA). As a result, even the private use of computer programs requires the permission of the rights holder, i.e. the granting of a licence.

ccc) Decoding of Computer Programs

134 Pursuant to Art. 21(1) CopA, the person who has the **right to use a computer program** (see above, D.IV.1.d)aa)) **may obtain the necessary information on the interfaces by decoding the program**. The use of the information obtained by decompiling is strictly limited; it may only be used for the development, maintenance and use of interoperable computer programs insofar as neither the normal exploitation of the decompiled program nor the legitimate interests of the owner of the rights in such program are unreasonably prejudiced (Art. 21(2) CopA) The limitation does not permit the user to develop and distribute a similar program.[238]

ddd) Backup Copies

135 The owner of a copy of a literary or artistic work may make a copy of such work to ensure its **preservation**. The original or the copy must be stored in an archive (Art. 24(1) CopA). A similar provision applies to computer programs. According to Art. 24(2) CopA, any person entitled to use a computer program may make a **backup copy** of it. While this provision was crucial when computer programs were acquired on physical media (e.g. floppy disk, CD or DVD), it has become largely irrelevant in times of cloud computing.

eee) Temporary Copies

136 Art. 24*a* CopA allows the making of temporary copies of a work if four conditions are met: (1) the copy must be **transient** or incidental; (2) the copying must represent an integral and

[238] EGLOFF, in: Barrelet/Egloff, Art. 21 URG para. 16.

essential part of a technological process, (3) the sole purpose of the copy is **to enable a transmission** of the work in a network between third parties by an intermediary or a lawful use of the work; (4) the copy has **no independent economic significance**.

137 The main purpose of this exception is to allow works to be **sent from one server to another via the internet**.[239] While the focus of the legislator was on enabling the distribution of literary and artistic works on the internet, this limitation also applies to the distribution of computer programs, e.g. the use of software as a service (SaaS) by the client of a cloud computing provider.

fff) Scientific Research

138 Art. 24*d*(1) CopA permits the reproduction of works for the purpose of scientific research, provided that the copying is due to the **use of a technical process** and that the works to be copied can be **lawfully accessed**. Once the scientific research is completed, the copies made in accordance with this limitation may be retained for archiving and backup purposes (Art. 24*d*(2) CopA). The limitation permits scientific research **for both non-commercial and commercial purposes**.[240]

139 However, Art. 24*d*(3) CopA **excludes computer programs from the exception for scientific research**. Data sets consisting of program data, source code or HTML code may not be reproduced for scientific purposes, even if they have been legally obtained or are publicly accessible.[241] **Regardless of whether the training and testing of an AI system may qualify as scientific research**[242], the use of program data, source code or HTML code for the training, validation and testing of an AI system capable of producing computer programs can hardly be permitted on the basis of the limitation for scientific research. As a result, computer programs may only be used to train, validate and test AI systems if the rights holder has granted a licence to do so. Arguably, the lack of a specific limitation for the use of computer programs for training, validating and testing AI systems is one of the **key deficiencies** of today's copyright regime (see below, E.V).

cc) Term of Protection

140 A work is protected by copyright when it is created, irrespective of whether it is fixed on a physical medium (Art. 29(1) CopA). For computer programs this protection expires **50 years after the death of the single author** (Art. 29(2) lit. a CopA) or of the last surviving co-author, respectively (Art. 30(1) lit. a CopA). These provisions implement the requirement of the Berne Convention, which provides for a minimum term of protection of the life of the author and 50 years after his/her death (Art. 7(1) BC).

2. Comparative Findings

a) Protected Subject Matter

141 In all analysed jurisdictions, the subject matter of protection is the literal expression of a computer program in the form of **source and object code** and not the underlying ideas. One of the main issues observed is to what extent the non-literal part of computer programs is protected. As the

---

[239] BBl 2006 3430; BGE 145 III 72 para. 2.3.2; REHBINDER et al., in: OFK, Art. 24*a* URG para. 7.

[240] BBl 2018 591, 629.

[241] REHBINDER et al., in: OFK, Art. 24*d* URG para. 10.

[242] See THOUVENIN/PICHT, sic! 2023, 516 et seqq.; CHERPILLOD, sic! 2023, 447; HARTMANN, sic! 2023, 164; ISLER, LSR 2022, 114 et seqq.

protection is limited to the source and object code, third parties can rewrite the code that fulfils the exact same function, thereby hollowing out the copyright.

142 **US courts** have developed an interesting and noteworthy example of a computer-program-specific application of copyright law. They adhere to the **"abstraction, filtration and comparison test"** that was developed in *Computer Associates Intern., Inc. v. Altai*. It is a three-step approach in which the court first breaks down the work into its levels of abstraction. The lowest level is the code, and the highest level of abstraction is the ultimate idea of the computer program. This should separate protectable code from unprotectable ideas. Second, the court dissects the program and filters out non-protectable elements. Those are elements based on efficiency (only one or two workable code options), elements based on external factors (e.g. hardware constraints) and elements taken from the public domain. Lastly, the remaining and thus copyrightable parts are compared to the allegedly infringing program, and it is decided whether the work was copied. In this final step, the court also considers the importance of the protected part of the code in relation to the entire computer program.[243] While this test does not really clarify the issue, it can **only be applied in a meaningful way to computer programs**, thus highlighting the fact that software does not fit within copyright.

b) Requirements for Protection

143 To be protected in all examined jurisdictions, works must be **original** (not a copy), contain a **minimal amount of creative effort** by the author and be fixed in a tangible medium. Between the examined jurisdictions, there is **no fundamental difference in the standard of creativity** for the expression in question; the differences are mainly semantic.[244]

144 It seems to be generally accepted in the analysed jurisdictions that literary and artistic **works are not protected by copyright if they have been autonomously generated by an AI system**. **Except for the UK**, all the examined **jurisdictions require the author of a work to be a human being**. Art. 9(3) CDPA states that if a work is computer generated, the author shall be taken to be the person who made the **necessary arrangements** for the creation of the work. Art. 178 CDPA specifies that "computer generated" refers to a work that is entirely generated by a computer so that there is no human author of the work. Hence, works created entirely by AI systems are protected under UK copyright law.[245]

145 However, there are **uncertainties** about the originality requirement[246] and it is unclear to whom the work should be attributed.[247] In one case the court applied Art. 9(3) CDPA, in which the **programmer and not the player of a videogame was identified as the person who made the necessary arrangements** for the appearance of visual effects. The court argued that in this case the programmer was responsible for all the skill and labour that was put into the display of the graphics.[248] Following the court's reasoning, one can infer that where an **AI system generates code based on a prompt provided by a user, the user would likely be considered the author** because they take the final step in creating the code.

---

[243] *Computer Associates Intern., Inc. v. Altai,* 982 F.2d 693 (1992), 706 et seqq.

[244] See also *Global Yellow Pages Ltd v Promedia Directories Pte Ltd*, 2 SLR 185 (2017), para. 27.

[245] UKIPO, AI & IP, paras. 19, 29, The UK Government is of the opinion that AI-generated works are currently protected under Art. 9(3) CDPA and decided to make no amendments to the law; BENTLY et al., 127 et seq.

[246] BENTLY et al. 127 et seq.; ATILLA, JIPLP 2024, 48 et seqq.; BOND/BLAIR, JIPLP 2019, 423.

[247] ATILLA, JIPLP 2024, 48 et seqq.

[248] *Nova Productions Limited v. Mazooma Games Limited and Others*, [2006] EWHC 24 (Ch), paras. 105 et seqq.

146 The **threshold of the necessary human input** to establish human authorship of a work varies between jurisdictions. While the USPTO sets a rather high bar and does not accept the argument that many iterations and prompts were used to create a work,[249] in China the iterative process and continuous adjustment of prompts by a human suffice for authorship.[250]

c) Acquisition of Rights, Ownership, Transferability

147 In all examined jurisdictions the **author is generally the first owner** of the copyright. As a consequence, there is a need to identify a human author to which the work can be attributed in all jurisdictions. This is even true for computer-generated works in the UK, where the author is the person who made the necessary arrangements for the creation of the work.

148 There are **special provisions for works created by employees** for which the employer automatically will become the owner of the copyright (e.g. under US law the company is even deemed to be the author). The copyright **can be transferred** in all examined jurisdictions. **Moral rights** are largely non-transferable, and in the US such rights are not even recognized for computer program works.

d) Effects of Protection

149 For all the examined jurisdictions, economic rights allow rights holders to **prohibit unauthorized use, reproduction, distribution or adaptation** of the work. There are specific **limitations** pertaining to computer programs in all the examined jurisdictions, mainly to allow for the decompilation of computer programs to ensure interoperability with other computer programs, and strictly for that use only.

150 The term of protection varies across jurisdictions and lasts **50 years from the year the work was made or 50 years after the death of the author, and up to 120 years after the first publication**. Considering the economic reality of the lifespan of computer technology, these terms are excessive.

e) Copyright Registration

151 In all examined jurisdictions, copyright arises with the creation of the expression. In the US, Japan and China, copyrights in a computer program can be registered and for the enforcement of certain rights such registration is a prerequisite. In the US, a lawsuit for infringement of copyright can only be filed if the right is registered (17 U.S.C. § 411).

152 In **Japan**, the transfer of a copyright and the establishment, transfer, alteration or expiration of a pledge on a copyright, as well as a restriction on its disposal, can **only be held against a third party after its registration** (Art. 77 JCA).

153 In **China registration is voluntary and should ease the proof** of ownership in disputes (Art. 12 CLC, Art. 7 RCSP).

154 The ratio of registration is **not public disclosure of source code**, and mainly serves as a tool for **proof of ownership**. None of the jurisdictions require registration of the entire source code; a relatively small part is sufficient, thus allowing rights holders not to publish the most valuable parts

---

[249] USCO, Zarya of the Dawn, 8.

[250] *Li Yunkai v. Liu Yuanchun,* Beijing Internet Court Civil Judgment Jing 0491 Min Chu No. 11279 (2023), 12 et seq.

of their computer program. In addition, some jurisdictions allow for measures to protect trade secrets contained in a computer program, e.g. the blocking of portions of the source code.

E. Deficiencies from a Legal-Economic Perspective

I. Starting Point: Overall Workability of the Current System – Strong Support for Reform

155 The stakeholder interaction and further fact finding in the course of this study present a twofold picture: overall, the market has **learned to somehow cope** with the current software protection regime and its shortcomings (see below). Nonetheless, most stakeholders we talked to take a **clearly positive view of reform tendencies**.

156 One main **driver** of this openness for reform are the **aforementioned shortcomings** of the current protection regime. Unfolding changes in market circumstances provide another driver, especially the **advent of widespread AI usage** with its far-reaching impact on the actual protection of current software rights and on the future protectability of AI-generated software.

157 The remainder of this chapter highlights **selected deficiencies**, and how they play out in the current **patent law and copyright law software protection prongs**. It also indicates examples for how our **Novel Approach addresses** these deficiencies. We focus on the following main deficiencies:

- provisions and rationales of the existing patent/copyright law framework as a bad fit for software
- current, frequent uncertainty over the availability of protection for software and computer-implemented inventions
- lack of transparency regarding the software IP stack and its protected subject matter
- multi-faceted protection thickets and access restrictions, unnecessary transaction costs
- AI software generation as a coffin nail for software copyrights

II. Bad Fit of Existing Provisions and Rationales – the "Sonderurheberrecht"

158 Just as foreign copyright laws, the Swiss Copyright Act **contains more provisions particular to software than to any other type of works**. This in itself indicates a bad fit between computer programs and copyright law. Taken together, the specific provisions on computer programs (namely Art. 2(3), Art. 10(3), Art. 12(2), Art. 13(4), Art. 17, Art. 19(4), Art. 21, Art. 24(2), Art. 24*d*(3), Art. 29(2)(a), Art. 30(1)(a), Art. 31(2)(a) and Art. 67(l) CopA) and the exclusion of the application of other provisions on computer programs (namely Art. 13(4), Art. 19(4) and Art. 24*d*(3) CopA) result in a differentiated legal regime that **could be described as a specific computer program copyright law ("Sonderurheberrecht")**.[251]

159 In addition, some of the fundamental concepts of copyright, namely the **concept of human authorship and the granting of moral rights, make little sense for computer programs**. The purpose of prototypical computer programs, the way they are used and the way they are produced is fundamentally different from the purpose, use and production of literary and artistic works. Computer programs provide solutions to specific tasks, while literary and artistic works are created to allow

[251] Similar: see STRAUB, Softwareschutz, para. 51; HILTY, Urheberrecht, para. 216.

authors to express their ideas and feelings and for other people to perceive them.[252] Computer programs are usually developed in large teams, using pre-existing code segments; the personality of the individual programmers is irrelevant as they do not strive to express their personality in the program but aim to provide a workable solution applying established standards and good coding practice. Accordingly, the focus on the individual author and its personality makes little sense for computer programs. The personality of the programmer is not violated by the fact that a computer program is published, that his/her name is not mentioned or that adaptations are made to the program. Although these rights have been granted by Swiss copyright law for more than 30 years, we are **not aware of a single case in which they have been enforced** and in our stakeholder exchanges **none of the experts affirmed the importance of moral rights** for computer programs. The irrelevance of moral rights for computer programs is further evidenced by the fact that US law does not provide for such rights at all (see above, D.IV.2.c)).

160 Many other provisions in copyright law are simply **irrelevant to computer programs**. This is the case for some economic rights (namely the right to recite, perform or present a work or to make it perceptible somewhere else [Art. 10(2)(d) CopA], the right to broadcast [Art. 10(2)(d) CopA], the right to rebroadcast [Art. 10(2)(e) CopA], the right to make a broadcast available [Art. 10(2)(f) CopA]), the right of access and exhibition (Art. 14 CopA), and protection against destruction (Art. 15 CopA). Most limitations are irrelevant to computer programs as well (see above, D.IV.1.d)bb)). In addition, application of the **exhaustion doctrine** to computer programs leads to overly complicated or even unsolvable problems as demonstrated by the famous "Used Soft" decision of the European Court of Justice[253] and the huge number of papers published which try to make sense of the exhaustion doctrine for computer programs.[254]

161 Finally, some general provisions of copyright law are **interpreted and applied differently** to computer programs than to literary and artistic works. The most prominent example used to be the protection requirement of the individual character, but the Federal Supreme Court has modified its approach in recent decisions in a way that allows application of similar criteria (namely neither banal nor routine creation) to literary and artistic works and computer programs (see above, D.IV.1.b)bb)).

162 The existence of specific provisions, the explicit exclusion of the application of other provisions, the irrelevance of entire concepts and the specific interpretation of certain general provisions for computer programs amount to a **specific regime for computer programs ("Sonderurheberrecht")** that has evolved over time. While this regime seems to work relatively well in practice, its emergence clearly indicates that the protection of computer programs could just as well be the subject matter of a specific IP right.

163 The **Novel Approach** would sever, or at least relax (in an improved version of the current framework), the ties to the traditional copyright and patent law legal framework and rationales, thereby permitting for an IP regime that is more specifically tailored to software particularities.

---

252 See SAMUELSON et al., Colum. L. Rev. 1994, 2347 et seqq.; STRAUB, Softwareschutz, para. 51.

253 CJEU, 3 July 2012, *Used Soft v. Oracle*, C-128/11.

254 To name just a few: HILTY, CR 2012, 626 et seqq.; SCHNEIDER/SPINDLER, CR 2012, 489 et seqq.; BÖTTCHER, 188 et seqq.

III. Uncertainty Over Availability of Protection in Current Settings – the COMVIK Example

164 Notwithstanding future AI developments, there is already considerable uncertainty over the protectability of a given piece of software. From a **copyright law perspective**, the increasing reliance by programmers on software-generating software does at least justify critical reflection on whether some of these software-generation processes overstep the human authorship boundary.

165 **Patent law's** approach to computer-implemented inventions provides a conspicuous example: today's software performs much more demanding and "technical" tasks than at the time the EPC and other pertinent fundamentals of software-patent law were conceptualized. Examples are computer-simulated measurements of physical conditions (temperature, density, etc.) or the virtual design and testing of product prototypes. AI systems will accelerate this tendency, as foreshadowed by "metaverse" concepts of differing viability. Although well intended and frequently well crafted, the attempts to reconcile historical legal categories with today's software realities seem increasingly byzantine and unconvincing. Even though its postulated lack of technical character was an axiom for the exclusion of software as such from patentability, the **COMVIK approach** has effectively removed this "hurdle" on the patentability level. At the level of novelty and inventive step, notions such as the "contribution" of non-technical features to a technical feature, "potential" technicality, "virtual"/"implied" technicality or the "sufficient specification" of a neural network's technical task clearly indicate the need to overcome the strict, technicality-based exclusion from patentability. However, as evidenced by the debates surrounding the EU's Software Patent Directive project (see annex, I.2), these notions also create a **high degree of legal uncertainty and make it hard for patent applicants to predict case law's future course**.[255] Intricacies regarding the program listings acceptable for a sufficient disclosure and description of the invention add to this effect.[256] The progress of AI in software generation will likely intensify these issues.

166 The complex state of the rules on CIIs generates **inefficiencies**, such as intransparency through insufficient code disclosure,[257] **high drafting costs, patent clusters** linking an underlying, software-implemented technology to various fields of "technical" application (e.g. products),[258] or excessively **broad process patents** which claim process protection for a computer program and its underlying algorithm, treating concrete implementations merely as specification examples that do not limit claim scope.[259] The former effect tends to **favour large patentees over MSMEs**, thereby potentially driving unwanted market concentration and a reduction in innovation dynamics.[260] The latter effect cuts against the goal to avoid thickets of low-quality patents.

167 As the **Novel Approach** does not start from a non-protection principle for software and its functionalities, which must then be reconciled with the technicality requirements for protectable inventions, it allows for a more efficient, more straightforward assessment of whether a given functional software claim deserves protection.

---

255 BALDUS, GRUR Int. 2021, 962; VALLONE, sic! 2019, 672.

256 See EPO Guidelines, F-IV, 3.9, F-II, 4.12.

257 SLOWINSKI, 353.

258 HEINRICH, in: PatG/EPÜ Kommentar, Art. 1 PatG/Arts. 52, 56, 57 EPÜ paras. 44 et seqq.; VALLONE, sic! 2019, 673 et seq.

259 SLOWINSKI, 353 seq.

260 See BALLARDINI, SCRIPTed 2009, 210 et seq.; BERGSTRA/KLINT, Sci. Comput. Program. 2007, 272.

IV. Lack of Transparency

168 Various elements of the current software protection landscape add up to an overall unsatisfactory level of transparency regarding existing software protection rights and their protected subject matters.

169 For example, Swiss **software copyright law lacks**, as the parallel rules in many other jurisdictions, a **register** for software copyrights and the protected source code. Regardless of how strictly one assumes the Berne Convention prohibits copyright registers, their frequent absence is a fact. However, our stakeholder exchanges have shown that a **register for computer programs could be helpful in various respects**. The lack of comprehensive registers for computer programs, and copyrights attached to them, impedes the identification of rights holders. This may hamper effective transacting over software copyrights and create liability risks for contracting parties (e.g. software developers) that guarantee – often *nolens volens* – the copyright compliance of their performance. Furthermore, our stakeholder exchanges confirmed that legal practice operates on a rather sweeping assumption according to which all code sequences of a certain (but undefined) length enjoy copyright protection. This assumption seems likely to generate a high error rate, and it results in part from the unavailability of comprehensive libraries (e.g. tied to a register) against which software plagiarism can be checked.

170 Two examples from **patent law** are code disclosure and DABUS issues. Regarding CIIs, it is often unclear whether the description of their software components effectively satisfies the **enabling disclosure requirement**.[261] Furthermore, uncertainties over how a human inventor requirement – confirmed for many patent laws in the course of the DABUS litigation – can be fulfilled for AI-generated software[262] leads companies to **disguise the true extent of non-human inventiveness** in the development of their software, as our stakeholder exchanges indicated. As an alternative, companies are likely to rely more on trade secrets and less on patents.[263] All these instances are worrying for a protection system that aims at generating invention publicity and, consequently, follow-on innovation. **Economic analysis** shows that incentivizing owners to **keep secret their technology hampers technology transfer**.[264]

171 A **Novel Approach** which enhances registration of software rights could thereby not only foster transparency. Register technology as it is available today (e.g. smart contracting technology interacting with a digital ledger) could also support **automated transactions** and render, the taking of standard copyright licences much more resource efficient.

172 Initial **economic analysis** indicates that the **incentivization of licensing transactions** will be a major justification for IP protection regarding AI systems and their output.[265] A software protection regime that fosters efficient licensing of AI-generated software draws additional legitimacy from this feature.

---

[261] See annex I.2 regarding lack of software patent transparency, in the context of the draft EU Software Patent Directive.

[262] Japanese case law has already denied the patentability of AI-generated inventions, see The Yomiuri Shimbun, Tokyo District Court Rules AI Cannot Be Issued Patents; Law Recognizes Only 'Natural Persons' as Inventors, 17 May 2024, available at: https://japannews.yomiuri.co.jp/society/crime-courts/20240517-186568/ (last accessed: 19 July 2024); Tokyo District Court, 16 May 2024, 2023 (Gyo-U) No. 5001, 23.

[263] THOUVENIN/PICHT, sic! 2023, 510.

[264] See for example HEGDE/LUO, Manag. Sci. 2018, 652; GANS et al., Manag. Sci. 2008, 988; DE RASSENFOSSE et al., Res. Policy 2016, 1326; MITRA-KAHN, 378.

[265] DE RASSENFOSSE et al., S. Cal. L. Rev. 2023, 107 seq.

V. Protection Thickets, Access Restrictions and Transaction Costs

173 In several respects, the software protection as designed by the patent and copyright law rules seems dysfunctionally strong.[266]

174 As described above (see above, C.I.1), the **sequential and cumulative nature of the software innovation** process tends to engender a **proliferation of software patents** creating protection thickets.

175 The **copyright term of protection** is **excessive** for computer programs, going far beyond their actual life- and usage-span. While this seems not to cause frequent, serious problems in practice, it has been dubbed "simply ridiculous" by stakeholders. To a slightly lesser extent, the same applies to the 20 years of **patent lifetime**, equally exceeding the life- and usage-span – and, for that matter, the incentivizing reward rationale – of the vast majority of CIIs.

176 Both **patent and copyright** law lack workable **limitations for AI training** purposes. In fact, one of the most important deficiencies of current copyright law is the lack of a limitation allowing the use of code for training AI systems. The Swiss Copyright Act only contains a limitation that allows the reproduction of works for the purpose of scientific research, and explicitly excludes such use for computer programs (see above, D.IV.1.d)bb)fff)). While there are important arguments as to why protected works may nevertheless be used for the training of AI systems,[267] this highly important issue should be clarified by introducing a specific limitation that permits the use of computer programs for the training of AI systems capable of generating computer programs. This would provide the much-needed legal certainty and ensure that the potential of AI systems to generate computer programs can be exploited. Ironically, the lack of reasonable (including cost) limitations may have contributed to patent and copyrights having often been disregarded in the training of AI systems. It would be much better to have **balanced, manageable limitations** in place, not only from the perspective of societal welfare, but also from that of rights holders.

177 Such a system could also contribute to a **smoother interaction** between the use and generation of software by AI systems and the **open-source movement**. The section on software protection rationales (see above, C.III) has explained the importance of open source as a factor in software markets. However, even open-source licences do not usually provide safe, innovation-friendly grounds for the AI training uses of open-source software.[268] It may even be that some parts of the open-source community react in an adverse, restrictive manner to the prospect that their software is being used to train commercial AI systems, as this use differs from the creative exchange between human programmers which constitutes a lead concept for the open-source movement. At the same time, there is no guarantee that software output of AI systems trained on open-source content will itself be available under an open-source licence.[269] In fact, our market research has not indicated a widespread nexus between commercial AI systems and open-source licensing.

---

[266] Regarding copyright thickets see ASAY, Emory L.J. 2017, 265.

[267] DE LA DURANTAYE, ZUM 2023, 657 et seqq.; see also MARMY-BRÄNDLI/OEHRI, sic! 2023, 664 et seq.

[268] VAUGHAN-NICHOLS, ZD Net 2023; for an overview of common open-source licenses in AI settings: MUÑOZ FERRANDIS/DUQUE LIZARRALDE, JIPITEC 2022, 224, 236 et seqq.

[269] The open-source initiative is pushing for open-source AI, see https://deepdive.opensource.org/wp-content/uploads/2023/02/Deep-Dive-AI-final-report.pd. Certain commercial AI providers make some of their systems available open-source, see e.g. https://www.ibm.com/blog/five-open-source-ai-tools-to-know/. However, AI providers that matter most on the market (e.g. OpenAI and Mistral) tend to reject or abandon open-source for their most valuable, cutting-edge systems, see https://sifted.eu/articles/mistral-microsoft-deal-controversy (all last accessed: 19 July 2024).

178 To end with a somewhat less conspicuous example, **co-ownership rules in patent and copyright law** can also contribute to protection thickets.[270] In particular, software is frequently written by teams of employees for the benefit of, and in exchange for payment by, their corporate employer. **Economic analysis** expects the **division of innovative labour to increase** in AI-heavy markets (e.g. between AI system-focused and system-output-focused innovators).[271] Extending co-entitlements in such situations adds an additional layer of complexity (e.g. transactions over corporate software). Furthermore, the **rights allocation to one (or several, possibly corporate) investor(s)** should gain in importance relative to the focus on a human author/inventor as the "natural" rights holder.[272] Mask works protection under Swiss law[273] exemplifies that it is not unheard of for IP law to sever the link between the attribution of an IP right and a human content generator. In contrast to copyright law, where protection is directed towards personal creation, the law for the protection of mask works was intended to be a means for investment protection, i.e. the protection of the industrial performance that is regularly associated with the development of a mask work.[274] Accordingly, ToA 3(1) stipulates that the exclusive rights originate with the manufacturer, which can be a natural or a legal person who has developed the mask work at his/her own expense and risk (ToA 3(2)). Moreover, from the outset the legislator anticipated that the manufacturer will in most cases be a legal entity.[275]

179 A **Novel Approach** could fight such protection thickets through a set of shorter protection terms, through a balanced system of AI-related limitations (possibly in exchange for compensation) and through entitlement allocation parameters more in sync with today's software-generation realities.

180 **Even an economic perspective that mainly fears AI systems as catalysts for software and IP rights thickets**[276] must acknowledge that an IP system which fosters software rights disclosure and (recompensed) limitations may well be preferable to a market situation framed by factual control over, and economic power based on, software silos.[277] The trajectory data-based markets have taken in the past 20 years and the heavy regulatory artillery now deployed to assail positions of control developed without the help of full-fledged IP rights corroborate this perception. In view of these reflections, it comes as no surprise that our **stakeholder exchanges indicated no tendency in favour of jettisoning IP protection** for software.

---

[270] See for example LEE, Am. U. Int'l L. Rev. 2016.

[271] DE RASSENFOSSE et al., S. Cal. L. Rev. 2023, 110.

[272] See for example ZECH, GRUR Int. 2019, 1147; Art. 9(3) UK CDPA attributing the right of the work created by a computer to "[…] the person by whom the arrangements necessary for the creation of the work are undertaken"; RAMALHO, 69 et seq., critical on this provision and underlining that, even absent a human author requirement, copyright law must perform the rights allocation task.

[273] Federal Act on the Protection of Topographies of Semiconductor Products of 9 October 1992 (SR 231.2).

[274] BBl 1989 III 477, 574.

[275] BBl 1989 III 477, 574.

[276] BESSEN/HUNT, 255 et seqq., suggesting that the expansion of property rights in the software industry can also result in a decline in the investment in R&D, inter alia because large companies prefer to manage their portfolios.

[277] See EVANS/LAYNE-FARRAR, Va. J.L. & Tech. 2004, paras. 66 et seqq.

VI. Demise of Software Copyright Protection Through AI Coding

181 Until quite recently, some of the above-mentioned (II.) deficiencies could be regarded as relatively theoretical flaws that have caused few significant problems in the IT industry (see above, E.I.) But this may well change with the growing importance of AI-generated computer programs. **Under current copyright law, AI-generated computer programs (or parts thereof) will not be protected by copyright for lack of a human author**.[278] In certain jurisdictions, even the patentability of AI-generated inventions seems at stake.[279] For the time being, there may still be a sufficient degree of human involvement to meet the requirement of intellectual creation and thus allow copyright protection to be granted.[280] However, this may change in a not-so-distant future, and there is also a risk that the human involvement in code writing will be overstated or even completely made up in court proceedings to support a claim for the alleged existence of copyright protection. In addition, companies that use AI tools to generate software – as is now standard practice – will no longer know whether the resulting programs are protected by copyright. Such legal uncertainty seems highly problematic, as the IT industry relies heavily on the assumption that computer programs are protected by copyright and can therefore only be used by third parties on the basis of a licence.

182 From an **economic perspective**, there are reflections on whether (software) **content generation will continue to merit incentivization through IP rights on the content output**, even though AI systems generate the content at irrelevant marginal cost.[281] Instead, the main investment worthy of protection and incentivization may take place at the level of content-generating AI systems.[282] Even from that perspective, however, loss of copyright protection for AI-generated software – in combination with an uncertain, and in any case limited availability of patent protection – constitutes a systemic challenge. This is because standard AI systems generating code will soon no longer be human-built but will themselves embody automated output of parent AI systems. Hence, such code-generating systems will likely not enjoy copyright protection or reliable patent protection.

183 Consequently, while there have always been good legal and economic reasons for creating an IP right tailored to computer programs and to the needs of programmers and the software industry, the growing importance of AI-generated computer programs **clearly calls for a fundamental change by creating a dedicated "Software Right"**.

184 Adding a dedicated Software Right to the IP family may prove to be politically difficult. While some adaptations seem inevitable to accommodate AI-generated computer programs, the **necessary adaptations** can also – in particular if the implementation of Software Rights seems currently unrealistic – be made **within the existing copyright law framework** by adjusting the interpretation of certain provisions (e.g. the requirement of human authorship, see above, D.IV.1.b)aa)), by excluding the application of others (e.g. moral rights, see above, D.IV.1.d)aa)), and by introducing some additional provisions, (namely limitations for bug fixing, for using computer programs for cyber

---

[278] See RAGOT et al., sic! 2019, 574; THOUVENIN/PICHT, sic! 2023, 511.

[279] Japanese case law has already denied the patentability of AI-generated inventions, see The Yomiuri Shimbun, Tokyo District Court Rules AI Cannot Be Issued Patents; Law Recognizes Only 'Natural Persons' as Inventors, 17 May 2024, available at: https://japannews.yomiuri.co.jp/society/crime-courts/20240517-186568/ (last accessed: 19 July 2024); Tokyo District Court, 16 May 2024, 2023 (Gyo-U) No. 5001, 23.

[280] See BGH, X ZB 5/22, 11 June 2024 – *DABUS*, para. 44, stating that in the current state of the art, there is at least one natural person that significantly contributes to the conception of an invention.

[281] See for example DE RASSENFOSSE et al., S. Cal. L. Rev. 2023, 105 seq.

[282] HILTY et al., 17 et seq.; YANISKY-RAVID, Mich. St. L. Rev. 2017, 700 et seqq.; YU, U. Pa. L. Rev. 2017, 1263.

security purposes, and for scientific research, see below, F.II.1.d)bb)). It is important that a dedicated limitation that allows for the use of computer programs for the training of AI systems be introduced.

185 Rather than just adapting the current provisions, a better way forward within existing copyright law might be to recognize that computer programs are fundamentally different from literary and artistic works by including a **dedicated section within copyright law** containing all provisions applicable to computer programs. This would not only reflect the fact that copyright laws already contain a specific legal regime for computer programs ("Sonderurheberrecht") but would also allow the drafting of provisions applicable to computer programs in a tailor-made manner. However, both approaches would remedy only the most important deficiencies.

186 A **Novel Approach** introducing a dedicated Software Right would allow for more fundamental changes. Such a right could be based, in particular, on notions of investment protection and innovation incentives, without the need to incorporate the dimension of personality rights. It would allow to overcome the limitations of copyright law when it comes to granting **protection for the underlying functionalities instead of only for sequences of code**. In many instances, these functionalities represent the actual value of a computer program. The indirect protection for functionalities that may be granted by copyright protection is often insufficient as it can be circumvented relatively easily, e.g. by using a different coding language. In fact, the need for an additional, functionalities-based protection has triggered the intense use of software/CII patenting[283] which comes with its own deficiencies (see above, E.III, IV and V).

[283] SLOWINSKI, 351.

F. Possible Solutions

I. Preliminary Remarks

187 The shortcomings of today's IP protection of software can be addressed in different ways. Recognising that computer programs do not fit well into either copyright, or patent law, an ideal solution would be to develop a **sui generis IP right for the protection of software**. The international treaties currently in force do not restrict the introduction of such a new Software Right, as they only harmonise the laws on existing IP rights. Ideally, a sui generis Software Right would **replace the existing protection provided by patent and copyright** law, thereby reducing the risk of overprotection. Patent law may be adapted to exclude (much more rigidly) the protection of computer programs, as no international treaty explicitly requires such protection. However, Art. 10(1) TRIPS Agreement requires all WTO member states to protect computer programs as literary works under the Berne Convention.

188 Since a revision of the TRIPS Agreement will not be possible for many years, a **sui generis Software Right** would have to **coexist with (at least) the current protection of software by copyright law**. Although this seems problematic at first, a suitable application of existing copyright law concepts, namely a demanding interpretation of the traditional human creation requirement which grants copyright protection only to computer programs truly created by human beings, could provide important incentives for rights holders to refrain from relying on copyrights in computer programs and to focus on a Software Right instead (see below, F.III.7).

189 Irrespective of the potential success of a Software Right that would coexist with the protection granted by patent and copyright, we need to **start developing suitable IP rights for the protection of software now**, in order to have convincing solutions at hand should an **opportunity to revise the TRIPS Agreement** arise. Surprisingly, our research did not reveal any recent attempt to develop such a right.[284] A key aim of this report is therefore to sketch some initial ideas (see below, F.III) and to (re-)launch a debate in the IP and software community.

190 In addition to the long-term project of developing a sui generis rights approach for the protection of software, the **current patent and copyright regimes should be revised** to address their shortcomings (see below, F.II). As opposed to creating a sui generis IP right, however, the scope and depth of such adaptations is limited by the given framework of international law, namely Art. 10(1) TRIPS Agreement and the provisions of the Berne Convention.

191 In sum, we suggest, as arguably the most workable approach, a three-stage implementation of our below proposals:

- Stage 1: **Improvements** to the current legal framework and **further research/discussion** on fleshing out a sui generis Software Right;
- Stage 2: **Implementation** of a sui generis Software Right (preferably through an EPC-style international treaty but possibly also in individual, pioneering jurisdictions that want to improve their IP framework for computer programs) and **coexistence** of the sui generis Software Right with the existing IP law framework. During such a coexistence phase, markets would assumingly gravitate towards using the sui generis right, aided by a restrictive practice regarding the patent and copyright protectability of computer programs;

[284] For earlier suggestions see SAMUELSON, Pat. & Licensing 1995; SAMUELSON et al., Comm. of the ACM 1996.

- Stage 3: **Predominant** use of the sui generis Software Right; possibly explicit removal of software protection from patent and copyright law in sync with an adaptation of the TRIPS Agreement.

II. Improvements to the Current System

1. Copyright Law

a) Leeway within International Treaties

192 The **scope for adapting national copyright laws under the TRIPS Agreement and the Berne Convention is unclear**, since the TRIPS Agreement only states that computer programs must be protected as literary works under the Berne Convention, and the Berne Convention does not mention computer programs at all. Given that most national copyright laws contain specific provisions for software, it is safe to assume that the contracting parties of the Berne Convention would have included specific provisions for computer programs if they had known that computer programs would be protected under the convention due to the referral in the TRIPS Agreement.

193 Since the Berne Convention does not contain such provisions, we must **assess the leeway for adapting national copyright laws with regard to the general copyright provisions** that were all drafted for literary and artistic works, and with regard to the specific provisions for specific types of works such as films, or dramatic and musical works. In some cases, the scope for adapting national laws will be very narrow, e.g. for the exclusive rights and potential formal protection requirements; in other cases, the scope will be rather broad, namely with regard to specific limitations to copyrights in computer programs.

b) Protected Subject Matter and Requirements for Protection

194 As mentioned above, Art. 10(1) TRIPS states that computer programs, whether in source code or in object code, shall be protected as literary works under the Berne Convention. National legislators must thus provide copyright protection for computer programs. Hence, there is **no scope** to adapt the subject matter of copyright laws **to exclude the protection** of some or all types of computer programs.

195 The Berne Convention does not limit copyright protection to literary and artistic works that were created by humans. Accordingly, national legislators could grant copyright protection to computer programs that were **autonomously produced by AI systems** as is already the case in the UK (see annex I.4.b)). However, in view of the key importance of the human contribution in Continental European jurisdictions, and in line with the prevailing view in legal literature (for Swiss law see above, D.IV.1.b)aa), for other Continental European jurisdictions see the references[285]), we do **not recommend changing** this requirement within the existing copyright law framework for any of the protected subject matters.

196 The **requirements for the protection** of literary and artistic works are not harmonised in the Berne Convention. Contracting states should use this leeway to develop specific requirements that are appropriate for computer programs. An approach relatively closely aligned with existing protection requirements could focus on whether the **available scope for developing the source code has been used to take creative and/or innovative code-writing decisions in such a way that the probability of an independent creation of (virtually) identical source code appears to be**

[285] For Germany, see THUM, in: Wandtke/Bullinger, § 7 UrhG paras. 16 et seqq.; BAUMANN, NJW 2023, 3676. For France, see VIVANT/BRUGUIERE, paras. 303 et seqq.

very small (see also below, F.III.3.a)). Such a requirement would also work for AI-generated source code and could be explicitly stated in national copyright laws. As established in patent law, the assessment of the requirement for protection should be made from the **perspective of an expert,** taking into account **standard coding practice**. This should allow to draw a line, convincing to the experts in the field, between computer programs that are protected and those that remain unprotected. As a result, unnecessarily complex programs may remain unprotected, while computer programs that are able to implement a functionality in a very simple and convincing way would be granted copyright protection.

c) Acquisition of Rights, Ownership, Transferability and Software Register

197 The Berne Convention does not harmonise the acquisition of copyrights, thereby allowing the **original acquisition of a copyright in a computer program by a legal person**. While Continental European copyright laws should maintain the requirement of the human creator (see above, D.IV.1.c) and D.IV.2.c)), they could allow for the original acquisition of copyrights in computer programs by legal persons. This approach would better reflect today's patterns of software generation (see above, E.V) and would provide a sensible solution for the acquisition of copyright in computer programs (co)generated by AI systems. Initial ownership would be vested in the natural or legal person that carries out the steps necessary to develop the computer program.

198 Nor is the **transferability** of copyrights harmonised by the Berne Convention. This allows for a full transfer and unlimited licensing of all rights granted for computer programs, at least assuming that moral rights are irrelevant for the protection of the personality and/or tacitly waived by programmers (see above, F.II.1.d)aa)d))

199 According to Art. 5(2) Berne Convention, the enjoyment and the exercise of the rights granted by the convention shall not be subject to any formality. This restriction **does not preclude the establishment of a register** which allows the owner of a copyright in a computer program to register the source code on a ledger, e.g. for evidentiary purposes or to provide a sound basis for the (automated) granting of licences. In fact, several member states of the WTO – e.g. the US, Japan and China – implemented copyright registers, including for computer programs (see above, D.IV.2.e)), and conditioned important modes of exercise of copyrights upon registration. These registers could be further developed to use technologies available today, e.g. for the (automated) granting of licences. We **recommend that national IP offices and/or WIPO start developing such a ledger**, preferably based on a joint effort and with the aim of providing a single ledger that can be used by all owners of copyrights in a computer program.

d) Effects of Protection

aa) Exclusive Rights

200 The Berne Convention (BC) contains a series of economic rights vested in the copyright owner. Some of them are granted for all literary and artistic works (e.g. the right of reproduction, Art. 9 BC), while others are granted only for specific types of works, such as dramatic and musical works (e.g. the right of public performance and of communication to the public of a performance, Art. 11 BC). While several economic rights are irrelevant for computer programs (e.g. the right of broadcasting [Art. 11bis BC], the right of public performance [Art. 11 BC] or the right of public recitation [Art. 11ter BC]), others are equally important for computer programs and literary and artistic works, namely the right of reproduction (Art. 9 BC). While the rights granted for specific types of works need not be granted for computer programs, all other **rights must also be granted for**

computer programs. Art. 10(1) TRIPS seems to be very clear in this respect, leaving no leeway for reducing the set of rights granted to owners of a copyright in a computer program.

201 The Berne Convention requires the contracting states to provide for two types of **moral rights**, namely the right to claim authorship of a work, and the right to object to any distortion, mutilation or other modification of or derogatory action in relation to the work which would be prejudicial to the author's honour or reputation (Art. 6bis BC). As opposed to the economic rights, the TRIPS Agreement does not oblige WTO member states to grant moral rights to the owners of a copyright in a computer program (Art. 9(1) TRIPS). Even the member states of the Berne Convention that are obliged to grant moral rights have considerable scope when interpreting and applying these rights. Given the nature of computer programs and the way in which they are developed, **courts may conclude that copyright owners in computer programs cannot prohibit distortion, mutilation or other modification of computer programs**, as these activities would not be detrimental to the honour or reputation of software developers. In addition, courts may adopt the view that **programmers waive the right to be named** if several programmers were involved in the development (see above, D.IV.1.c)). At the very least such a tacit waiver could also be assumed if the program was developed on behalf of a third party, e.g. an employer or a contractor. Given that the Berne Convention contains different provisions for different types of literary and artistic works, and given that computer programs were not considered when Art. 6bis BC was introduced, it can even be assumed that the Berne Convention would not limit the ability of national legislators to **refrain from granting moral rights to programmers**.

bb) Limitations

202 Whether and to what extent the economic rights can be limited by national legislators by introducing exceptions and limitations in their national copyright laws is subject to the **three-step test**. Even if this test is only provided for in the Berne Convention for limitations of the right of reproduction (Art. 9(2) BC), it applies today to limitations to all exclusive rights due to its general formulation in Art. 13 TRIPS and the corresponding provisions in Art. 10 WCT, Art. 16 WPPT and Art. 13 Beijing Treaty. This is recognized both internationally[286] and nationally.[287] Art. 13 TRIPS formulates the three-step test as follows: "Members shall confine limitations or exceptions to exclusive rights to certain special cases which do not conflict with a normal exploitation of the work and do not unreasonably prejudice the legitimate interests of the right holder". The three-step test is formulated in a very open way and designed as a rule that gives national legislators a **very wide scope to introduce limitations and exceptions** in their copyright laws.[288] Accordingly, there is ample leeway for national legislators to introduce specific limitations to copyrights in computer programs.

203 Many copyright laws contain specific limitations for computer programs (see above, D.IV.1.d)bb) and D.IV.2.d)). The European legislator harmonised the exceptions and limitations in **Art. 5 of the Directive 2009/24/EC** on the legal protection of computer programs. Accordingly, European copyright laws allow for (1) the use of the computer program by the lawful acquirer in accordance with its intended purpose, including for error correction (Art. 5(1) Dir. 2009/24); (2) the

[286] RICKETSON/GINSBURG, para. 13.125; BRAND, in: Busche et al., Art. 9 TRIPS para. 57; GEIGER et al., GRUR Int. 2008, 822.

[287] BGE 133 III 473 para. 6.1; EGLOFF, in: Barrelet/Egloff, Pre. Art. 19–28. para. 5; HILTY, Urheberrecht, para. 437; CHERPILLOD, SIWR II/1, para. 745; RUEDIN, in: Commentaire romand, Pre. Art. 19–28 URG para. 5.

[288] SENFTLEBEN, GRUR Int. 2004, 202; GEIGER et al., PIJIP 2013, 12; critical of this scope: REIMER/ULMER, GRUR Int. 1967, 444.

making of a backup copy by a person having a right to use the computer program (Art. 5(2) Dir. 2009/24); (3) the right to observe, study or test the functioning of the program in order to determine the ideas and principles which underlie any element of the program (Art. 5(3) Dir. 2009/24); and (4) the right to carry out decompilation (Art. 6 Dir. 2009/24) (see annex I.3).

204 The **Swiss CopA** only contains two specific limitations for computer programs, namely the **limitation for decompilation** (Art. 21 CopA) and for the **making of a backup copy** (Art. 24(2) CopA) (see above, D.IV.1.d)bb)). At least these limitations should apply in the future.

205 However, given the ample leeway for national legislators to introduce specific limitations to copyrights in computer programs based on the three-step test, taking into account the limitations provided for in the EU and considering additional needs voiced in our stakeholder exchanges, we recommend that the Swiss legislator **include additional limitations** for the use of computer programs, namely:

- **Limitation for error correction (bug fixing):** As mentioned above, Swiss copyright law does not contain a limitation for bug fixing but some scholars argue that such a limitation nevertheless exists (see above, D.IV.1.d)aa)). Given the practical importance of bug fixing, we recommend introducing an explicit limitation that allows for the use of computer programs for this purpose. Such amendment would be in line with European law and was unanimously supported in our stakeholder exchanges. Given that many rights holders will prefer to fix bugs in their computer programs themselves, the limitation should only be granted if the rights holder has failed, upon notification, to fix the bug in a timely manner. As with decompilation, the limitation for bug fixing could be structured as a compulsory licence with an information duty, compelling rights holders to provide third parties with the information necessary to fix a bug.
- **Limitation for analysing and adapting computer programs for the purpose of information technology (IT) security:** IT security is one of the key vulnerabilities of today's digital societies, and achieving a satisfying level of security is technically challenging and extremely costly. In many cases, attacks are based on the exploitation of vulnerabilities of computer programs. Legislators should facilitate all activities needed to strengthen security. Therefore, we recommend including an explicit limitation that provides for analysis and adaptation of computer programs for security purposes. This limitation could be introduced in a separate provision, or it could be combined with the limitation for bug fixing, subject to the same conditions (i.e. the failure of the rights holder to remedy the problem) and structured in the same way as the latter (i.e. as a compulsory licence with an information duty).
- **Limitation for the use of computer programs for scientific purposes:** The current limitation for the use of works for scientific purposes (Art. 24*d* CoPA) explicitly excludes, in its para. 3, the use of computer programs (see above, D.IV.1.d)bb)fff)). The reason for this exception remains unclear, as the legislator has not provided any substantive explanation.[289] Just as literary or artistic works, computer programs may also be the subject of research; allowing the necessary reproductions to be made for the application of advanced research techniques to computer programs would facilitate such research. This adaptation of today's copyright regime for computer programs could be implemented by simply deleting para. 3 of Art. 24*d* CopA.
- **Limitation for the use of computer programs for training, validating and testing AI systems:** While it seems clear that such a use **needs to be made possible, in an effective and efficient manner**, there are differing **design options** and the need for a thorough discourse

[289] BBl 2018 591, 629.

regarding several aspects of the limitation. These include whether it is granted free of charge or against payment of an adequate compensation; whether it offers an opt-out possibility, allowing rights holders, as a first step, to prohibit others from using their computer programs for the training, validation and testing of AI systems while also serving as a possible starting point for licensing negotiations; whether such opt-out should include a FRAND access granting obligation in case opt-out-induced licence negotiations fail; and how to organise royalty collection and enforcement. From today's perspective, a **limitation** with an **opt-out** mechanism in a **machine-readable format** seems to a promising approach to balance the interests at stake. This approach would reduce transaction costs for users of AI systems while allowing rights holders to prohibit the use of their code for the training, validation and testing of AI systems, or to allow such use against payment of an individually agreed royalty.

- As shown by the *Microsoft* decision of the European Court of First Instance,[290] today's **limitations for decompilation** in copyright laws are **unable to fully meet the intended goal**, as third parties may not always be able to attain the necessary interoperability information. In its decision, the Court of First Instance had to remedy this problem by applying competition law. However, the structural potential of IP rights to distort competition should, where possible, be dealt with by IP law, as this allows to coherently tailor solutions into the overall IP law framework. Furthermore, a stronger reliance on IP law enforcement mechanisms reduces the need to rely on the enforcement by competition authorities, whose resource constraints and expertise may not be suited to ensure comprehensive, timely and cost-efficient protection of competition in this area. Instead of a blunt statutory exemption that merely allows third parties to decompile computer programs for interoperability purposes, the problem would be better addressed by introducing a more specific **limitation, including a flexible information and support obligation that** requires rights holders to make interoperability work effectively. As it would compel rights holders and access seekers to **cooperate**, this approach could also foster **case-specific solutions** (e.g. suitable means to protect trade secrets) beyond the granularity level of a statutory provision.

cc) Term of Protection

206 The **leeway for adapting the term of protection for computer programs is somewhat unclear**. While the Berne Convention provides for a minimum term of protection of the life of the author and 50 years after his/her death (Art. 7(1) BC), the TRIPS Agreement acknowledges that the term of protection may be calculated on a basis other than the life of a natural person. In such a case, the term of protection shall be no less than 50 years from the end of the calendar year of authorized publication or, failing such authorized publication, within 50 years from the end of the calendar year of the making of the work (Art. 12 TRIPS).

207 Considering that TRIPS was drafted with a view to providing copyright protection for computer programs, it could be argued that Art. 12 TRIPS should be read as a provision allowing the term of protection for computer programs to be calculated on a basis other than the life of a natural person. In fact, the Berne Convention contains a similar provision for cinematographic works, allowing countries to provide that the term of protection for such works shall expire 50 years after the work has been made available to the public with the consent of the author or, failing such an event, within 50 years form the making of such work (Art. 7(2) BC). If one agrees with this interpretation, it seems possible to **reduce the term of protection for computer programs in the CopA to a period of 50 years after the commercialization of a computer program**, or failing such placing

[290] CFI (Grand Chamber), 17 September 2007, *Microsoft Corp. v Commission of the European Communities*, T-201/04.

on the market, 50 years after the development of the program. This interpretation is supported by the fact that **similar solutions have been implemented in other jurisdictions** that are also member states of the WTO. In the UK, computer-generated works, including computer programs, are protected for 50 years from the calendar year the work was made (see annex I.4.b)). In China the copyright protection of software expires 50 years after the first publication if the software is a work made for hire, and the copyright thus vests in the legal entity or the unincorporated organisation under which it was created (see annex I.7.b)).

208 Given that even this reduced term of protection would be excessively long, such an adaptation of the CopA would not completely solve this problem. However, it would be a relatively simple way to **at least mitigate** the problem of the excessively long term of protection for computer programs within the given framework of the TRIPS Agreement and the Berne Convention. In addition, calculating the term of protection on the basis of commercialization of a computer program would better reflect the fact that the individual programmer (and his or her lifespan) is hardly relevant in today's software industry and would allow a seamless application of the adapted provision to computer programs autonomously generated by AI.

e) Implementation

209 The improvements to the current copyright regime for computer programs could be implemented in two different ways:

- **Adaptation only:** the existing provisions could be adapted (i.e. deleting Art. 24d(3) CopA) and the novel provisions (e.g. the limitation for the training of AI systems) could be implemented at a suitable place within existing copyright acts.
- **Specific regime:** legislators could create a specific chapter within copyright acts that contains all, and exclusively, the provisions that apply to computer programs.The existing specific regime for computer programs ("Sonderurheberrecht") could thus be made explicit. Most importantly, this approach would make possible the creation of a tailor-made copyright regime for computer programs. Copyright acts as a whole would not necessarily become shorter, as some issues would have to be regulated for both literary or artistic works and computer programs (e.g. derivative works). The novel chapter on computer programs itself could be relatively short, and all computer-program-specific provisions within the chapter on literary or artistic works could be deleted.

210 While both approaches would allow for similar improvements to the current system, the creation of a **specific regime for computer programs** within copyright laws seems **preferable.** This approach could provide important clarifications by avoiding future confusion as to the application of traditional copyright doctrines to computer programs, as still is the case for the exhaustion doctrine (see above, E.II). This would enhance legal certainty. In addition, a specific regime could be a useful intermediate step towards the development and implementation of a sui generis Software Right (see below, F.III).

2. Patent Law

a) Functionality Instead of Technical Character

211 As described above, the "technical character" criterion generates much complexity and legal uncertainty, especially for frontier technologies such as complex simulations or neural networks. Instead of pursuing the current case law path which gradually, and not always coherently, relaxes the

technical character requirement, patent law could take a bolder step and **accept software-integrating patent claims** as long as they claim – as a whole, and without distinguishing between technical and non-technical features – a **functionality** in the sense described below (see below, F.III.3.b)

212 This would require giving up substantial parts of the COMVIK approach and thus a change of case law and patent examination guidelines. A **modification of Art. 52(2)(c) EPC may facilitate the shift but seems not an unavoidable necessity** because the "as such" wording could be read to exclude software which does not form part of a "functionality" claim in the sense used here, just as it is read today as referring only to software not combined with sufficient technical claims or features. To **illustrate** this with the **example** of **pedestrian simulations** (see also below, F.III.2.b)), a revised reading of Art. 52(2)(c) EPC would not permit claiming exclusivity for any software simulating a flow of people. Rather, protection could only be claimed for a specified software that optimizes the simulation of such flow in a specified manner, based on a specified set of parameters (e.g. width of passage, average walking speed, typical reactions to signs or colours). For patents granted under laws which, like the Swiss Patent Act, do not contain an explicit "as such" exclusion, the change to a more functional approach would be even easier.

213 Admittedly, a more functional approach would **tend to expand patent protectability for computer programs**, and thus call for a **counterbalancing** that mitigates harmful effects on innovation and competition (patent thickets, overly broad protection, etc.). While difficult to codify and ensure at the abstract level of statutory language, **demanding thresholds for novelty and inventive step** can be of crucial importance in this respect, in addition to an appropriate **limitations regime** (see below, F.II.2.c)).

b) Ownership and Transferability

214 For patent law, two suggestions go to the nexus between inventive activity and patent ownership.

215 Particularly[291] with regard to software, today's intensive use of automated code generation and the prevalence of software development by corporate teams (see also above, C.I.1 on sequential and cumulative invention processes in the software sector) have shifted market realities, at least in this sector, far from patent law's traditional concept of a human inventor to whom the innovative feat can be attributed so clearly as to justify his or her individual patent ownership. As we have argued elsewhere,[292] allowing for **"corporate patents"** (i.e. patents initially acquired by legal persons)[293] would contribute to a patent system that is more efficient and better adapted to today's innovation processes in the software industry.

---

[291] For a more general concept to make the human inventor/patent owner nexus more flexible, see ABBOTT, B.C. L. Rev. 2016, 1103 et seq.; GAJECK/SCHEIBE, RDi 2023, 413; KONERTZ/SCHÖNHOF, ZGE 2019, 403; NÄGERL et al., GRUR 2019, 340 et seq.; THOUVENIN/PICHT, sic! 2023, 509 et seq.

[292] THOUVENIN/PICHT, sic! 2023, 514 et seqq.

[293] In some settings, such acquisition may have to be combined with a monetary compensation for humans participating in the inventive process. Note that wages or other employment benefits calculated to compensate for innovative activity will usually obviate the need for additional compensation.

216 Corporate patent acquisition has been advocated by some scholars in the past.[294] At **least in the case of inventions generated solely by AI systems** (and assuming patentability for such inventions),[295] **sceptical views[296] are no longer convincing**. Inventions without a human inventor will be made in the foreseeable future, where the question of the patent allocation right can no longer be answered by referring to the human inventor(s). Since there is no human inventor and since AI systems cannot be the rights owner because of their lack of legal capacity, only the legal entity which is the owner of the company that has generated a patentable invention through the use of an AI system can reasonably be considered as the original owner of the right to the patent.[297]

217 As a particularly worthwhile reaction to the role of legal entities in today's innovation processes, the **legislator** should consider **allowing the original acquisition** of the right to a patent covering an AI-generated invention by a **legal entity that organised and financed the use of the AI system in research and development**.

218 The same goes for **permitting patent applications to designate an AI system as the (co)inventor**, while allocating ownership to a resulting patent based on parameters other than solely inventive contribution (on such parameters, see below, F.III.4.b)): The question of whether patent law can and should recognize AI systems as inventors if such systems generate an otherwise patentable technical teaching without an inventive contribution by a human is among the most intensely discussed questions at the interface of AI and IP. In addition to the academic debate,[298] the complex DABUS legal dispute plays a central role. It necessitates the most important patent law systems to determine whether they permit the inventorship of AI systems.[299] To date, patent applications based on the alleged inventions of DABUS have been rejected in the vast majority of jurisdictions. The rejecting patent offices or courts usually and correctly take the view that the applicable standards of patent law are geared towards human inventorship and not towards machines as inventors. At least one court has even taken the position that AI-generated inventions are not patentable at all.[300]

219 *De lege ferenda*, however, there are **important reasons why patent applications should openly and transparently describe the role that AI systems have played in the invention**

---

294 See for example STAEHELIN, in: ZK, Art. 332 OR para. 2; ANDERMATT, SJZ 2008, 286 et seqq.

295 On Japanese case law opposing the patentability of AI inventions, see The Yomiuri Shimbun, Tokyo District Court Rules AI Cannot Be Issued Patents; Law Recognizes Only 'Natural Persons' as Inventors, 17 May 2024, available at: https://japannews.yomiuri.co.jp/society/crime-courts/20240517-186568/ (last accessed: 19 July 2024); as a very recent case law example in favor of patentability, see however BGH, X ZB 5/22, 11 June 2024 – *DABUS*, para. 44.

296 For traditional settings see for example BREMI, in: SHK, Art. 3 PatG paras. 5 et seqq.; ZUBERBÜHLER, 104 et seq. For Germany: BGH, Ia ZR 110/64, 5 May 1966 – *Spanplatten*; X ZR 54/67, 10 November 1970 – *Wildverbissverhinderung*; KEUKENSCHRIJVER, in: Busse/Keukenschrijver, Art. 6 PatG paras. 13 et seqq.

297 More cautious, though, ANN, § 1 paras. 25 et seqq.; § 19 paras. 17 et seqq.

298 See for example BONADIO et al., Int. Prop. Q. 2021; SHEMTOV, inventorship; GAJECK/SCHEIBE, RDi 2023; KRAUSEN, GRUR 2023; DORNIS, GRUR Patent 2023.

299 See for example *Thaler v. Vidal*, 43 F.4th 1207 (Fed. Cir. 2022), the petition for writ of certiorari was denied: *Thaler v. Vidal*, 143 S. Ct. 1783 (2023); EPO, Grounds for decisions of 27 January 2020 on EP18275163.6 and EP18275174.3; 11 W (pat) 5/21 (*Food Container*), 18 W (pat) 28/20 (*Neural Flame*) and 12 W (pat) 21/20 (*Fractal Container*); LBA, 21 December 2021, J 0008/20 – *Designation of inventor/DABUS*; *Thaler v Comptroller*, UK High Court: [2020] EWHC 2412 (Pat) and Court of Appeal: [2021] EWCA Civ 1374; *Thaler v Commissioner of Patents* [2021] FCA 879 and *Commissioner of Patents v Thaler* [2022] FCAFC 62; Tokyo District Court, 16 May 2024, 2023 (Gyo-U) No. 5001; BGH, X ZB 5/22, 11 June 2024 – *DABUS*.

300 The Yomiuri Shimbun, Tokyo District Court Rules AI Cannot Be Issued Patents; Law Recognizes Only 'Natural Persons' as Inventors, 17 May 2024, available at: https://japannews.yomiuri.co.jp/society/crime-courts/20240517-186568/ (last accessed: 19 July 2024).

process. The need to assess whether the human contribution to an invention, compared to the contribution of an AI system, is sufficient to establish the inventorship of a natural person is detrimental to legal certainty and ties up the resources of patent offices. One of the most important objectives of the patent system is to inform the (specialist) public about technical progress and thereby promote further research and development. If patent applicants conceal the actual relationship between the contribution of humans and AI systems to an invention for fear of their application being rejected, this function suffers. The effect is exacerbated if AI-generated inventions are not submitted for patent protection but are kept secret, out of the concern that the use of inventive AI systems may prevent patenting.

220 These and other advantages of transparency on the inventive contribution of AI systems have **led the courts to look for creative solutions within the framework of current patent law**. The German Federal Court of Justice, the German Federal Patent Court and the Boards of Appeal of the EPO seem to accept a kind of human inventorship in representation for an innovative AI system. According to this concept, a natural person must still be named as the inventor in the application, though it is possible to declare in the description that the inventive acts were performed by an AI system[301] or at least that the designated human inventor largely relied on an AI system[302]. This concept of inventorship by representation is somewhat formalistic but it may well serve as an acceptable interim solution until patent laws adapts to AI invention settings.

221 Ultimately, if an **AI system has made an invention** without inventive human intervention, **this ought to be stated in the patent application**. The AI system should be named as the inventor. In addition, a **natural or legal person who is entitled to the patent should be named**. Until the provisions of Patent Acts are amended, natural persons should be named as "inventor representatives" in the patent application and registered in the patent register. However, this should require disclosure that the persons registered as inventors have not actually made the invention themselves. This disclosure should be included in the description. It should also contain sufficiently precise information on the nature and scope of the inventive contribution of the AI system.

c) Limitations

222 A functional approach should establish protection limitations akin to those for the Functionality-Level Software Right under our Novel Approach (see below, F.III.5.a)).

223 The current limitations regime should be amended by introducing, in particular:

- a **reverse engineering/decompilation right** not only for research purposes but also for bug fixing and, possibly, for (cyber) security purposes, to the extent the reverse engineering/decompilation and the use of the information gained would infringe a software patent at all (see above, F.II.1.d)bb);
- a limitation for the use of computer programs for **training, validating and testing AI systems** (see above, F.II.1.d)bb);

---

[301] See for example LBA, 21 December 2021, J 0008/20 – *Designation of inventor/DABUS* and LBA, 21 December 2021, J 0008/20 – *Designation of inventor/DABUS II*, 4.3.7; BPatG, 11 W (pat) 5/21, 11 November 2021, II.2.c).

[302] BGH, X ZB 5/22, 11 June 2024, paras. 52 ff. – *DABUS*, rejecting however the possibility to disclose an AI as the true inventor in the description.

- (remunerated) patent-law-based limitations for settings where hitherto access was effectuated by **competition law** (see above, F.II.1.d)bb)).[303]

d) Term of Protection

224 Even a reform that stays within the patent system framework should aim for **shorter actual protection** periods. The international legal situation (Art. 33 TRIPS in particular) would render a shortening of software/CII-specific protection periods difficult. **Renewal fee increases** could, however, encourage earlier patent waivers.

e) Consistency with International Law

225 We believe that the above suggestions are **compatible** with the software protection obligations **under international patent law**, especially with the TRIPS Agreement. In this respect we **refer to section G.II.2.a)** on international law.

III. A Novel Software Right

1. The Case for and Principle of a Software Right

226 The analysis of the current legal situation and the inputs we received from stakeholders **suggest going beyond the modifications described heretofore,** and introducing instead a novel IP right for software **("Software Right")**. Except for a few representatives of very large digital companies, we have observed broad support for such far-reaching changes to the current protection system.

227 Our analysis has clarified that the **existing regime**, with its coexisting copyright and patent law, has evolved historically and is **not the result of a coherent concept** for the adequate protection of software.

228 **Copyright** protection can often be easily circumvented by using a different programming language or by modifying the code just enough to escape the limited scope of protection. An excessive protection period, the granting of irrelevant economic and moral rights, the lack of transparency where there is no register and insufficient limitations favour adverse effects of existing software copyrights on innovation and market transactions. The multitude of software-specific provisions in copyright laws demonstrates that the copyright system squares badly with the particularities of software markets.

229 The **patent system** does not adequately protect software either. Patents relating to software emerged as a result of the lack of functionality protection within the copyright system. A rigid reading of the "technical character" threshold has not proven workable given the present-day prowess of computer programs. As case law attempts to gradually relax the threshold and adapt it to the swift pace of technological progress, it creates inconsistencies and legal uncertainty. The need for successful patent applications to navigate this complex terrain tends to disfavour single person or MSME patentees. Furthermore, the term of patent protection for software and the limitations to it suffer from inadequacies, similar to those under copyright law.

230 **Both patent and copyright** protection for software are tied **too closely to a human inventor/creator**, instead of emphasizing the software's innovative contribution. The increasingly autonomous generation of code by AI systems will further complicate the definition and detection of

[303] For example, a refusal to license by a market dominant rights holder that violates Art. 7 Swiss Cartel Act.

a creative human contribution sufficient to ensure copyright[304] protectability. A human creator/inventor focus may lead to undesirable incentives to claim a relevant amount of human contribution, even when it is lacking. Ultimately, the focus on a human creator may well render copyrights irrelevant as a means for software protection if software is to be produced fully autonomously by AI systems.

231 These findings suggest that a new, sui generis IP right for software is **not only possible but makes sense**. Such an IP right should provide for an easily obtainable source code protection, and for the protection of specific and innovative software functionalities, both independent from a rigid human inventor/creator requirement.

232 While international law does not restrict the introduction and design of a sui generis IP right (see above, D.II), the **design of such a right must take the market realities into account**. As the IT industry relies heavily on the existing patent and copyright protection, together with a dense, established network of related licensing contracts, a novel IP right can only be introduced if it accommodates these market realities. Accordingly, a novel right must be a workable substitute for patent and copyright protection, including transferability and open-source licensing.

233 Consequently, we propose a two-tiered sui generis Software Right. Such a right would protect the central innovation value aspects of software: the concrete formulation of the source code, its translation into binary code and the functionalities which computer programs are able to perform. The first tier would be a **Code-Level Software Right ("Code SR")**; the second would be a **Functionality-Level Software Right ("Functionality SR").** The two rights would protect various aspects of software and therefore differ in terms of subject matter and protection requirements. They would, however, be largely parallel in terms of rights granted, limitations and term of protection.

234 Our exchanges have shown that different **stakeholders** – e.g. start-ups, Big Tech, software developers and software users – have **different but not fundamentally contradictory needs**. Our Software Right would fulfil, inter alia, the following market needs and stakeholder *petita*:

- Informal, **automatic protection upon the generation of eligible software**, albeit limited to the protection of the source code (including the binary code) and a shorter protection period. **Stakeholders** considered it important that software is informally and effortlessly protected as soon as the source code has reached a certain length and complexity. This explains why most software producing companies attach importance to copyright protection, in spite of its flaws in other respects.
- Possibility to **protect** software – upon registration and fulfilment of protection requirements – not only in terms of the concrete formulation of the source code but also **in terms of its functionality**. **Stakeholders** pointed out that the value of software often lies not primarily in the formulation of the source code but in the functionality that a program provides. Today, this aspect is, though in a somewhat makeshift manner, covered by patent law if the requirement of technicality is fulfilled.[305] An alternative system should also provide for a reasonable form of functionality protection. Removing functional protectability by software patents without offering a replacement solution would not seem viable.
- **Severance from the human author/inventor principle**, thus allowing for the protection of software irrespective of whether it was developed with the use of AI and regardless of whether a

[304] The Yomiuri Shimbun, Tokyo District Court Rules AI Cannot Be Issued Patents; Law Recognizes Only 'Natural Persons' as Inventors, 17 May 2024, available at: https://japannews.yomiuri.co.jp/society/crime-courts/20240517-186568/ (last accessed: 19 July 2024).

[305] This functional protection by patents already complies with Art. 9(2) TRIPS.

sufficient degree of human creativity or inventorship can be identified. As widespread AI generation of software seems imminent, **stakeholders** desire a protection system that copes with this development in a non-defensive manner.

- **Allocation and ownership rules adequate to market realities**, such as (default) corporate ownership and co-ownership rules which would favour valid bona fide transactions. For **stakeholders**, it is imperative that the protection and transaction costs remain limited. This applies especially to MSMEs which are more constrained in resources and therefore more burdened by complex filing and transaction frameworks.
- Transparency through **registration**, with an effective registration system based on a digital register (**SR Ledger**) that would permit the storage of large volumes of code and automated transactions. Many **stakeholders** are interested in the documentation of existing protection and in increased transparency regarding rights holders through a register for software rights. Such a register would have beneficial effects for their licensing, collateralized debt and M&A transactions. Digital register technology facilitates the management of a potentially heavy register containing very large amounts of code. Stakeholders appreciate a digitally interactive register with these features, including as a blueprint for other areas of IP law.
- Comprehensive **economic rights,** which allow rights holders to **control any use** of the computer program (including private use), but which confer **no moral rights.**
- **Full transferability** of both the Code SR and the Functionality SR.
- **Tailor-made limitations**, including (recompensed) limitations for training, validating and testing AI systems. For many **stakeholders** it seems desirable to introduce a (licensing) system that furthers access to software, effectively and on fair terms, where such access is crucial for competition and innovation. Stakeholders emphasised that in the development of new computer programs the use of functions/parts of existing computer programs is often very beneficial. **Stakeholders urged** that the issue of software use for training, validating and testing AI systems capable of developing software needs to be addressed in a pro-innovative, balanced manner.
- A **non-excessive term of protection** which would allow for **extensions** subject to registration and payment of a fee.
- **Uniform, cross-jurisdictional SR-protection,** as opposed to national rights fragmentation under the territoriality principle, because of a single legal framework of reference (possibly enshrined in an EPC-style international treaty in the future); strong transactional efficiencies therefrom.

2. Protected Subject Matter

a) Code-Level Software Right

235 The protected subject matter of the Code SR would be the **source code in all its forms**, including in binary code. Given that the Code SR arises without entry in a register (see below, F.III.4.a)aa)), its subject matter cannot be defined in an abstract manner via the register entry and must therefore be identical to the source code as it is made available on the market or otherwise made public.

236 As is the case in other IP laws, namely patent, design and trademark law, appropriate **exclusions** would have to prevent the granting of Code SRs for software that **violates fundamental societal interests**.[306]

b) Functionality-Level Software Right

237 The key value and innovative contribution of software usually lie in its ability to perform certain functionalities. **Software patents already protect functional claims** when patent protection requirements are met. Other types of IP rights grant functionality protection for software as well, such as the Austrian utility model.[307] Software patents and similar IP protection have developed because code-focused copyright protection has proven insufficient for market needs. There are good reasons for a novel Software Right to grant functionality-based protection.

238 We are aware that the **protection of functionalities may raise issues regarding over-protection and blocking positions** harmful to competition and innovation. However, we believe these risks **can be mitigated** by appropriate tailoring of the **protection requirements (including the notion of "functionality"), the scope of protection and protection limitations,** as set out in subsequent paragraphs. As a fallback solution, **competition law** would always remain applicable to also prevent abusive uses.

239 Working out an appropriate **concept for protectable functionalities** will require **further research and discussions**. Arguably, the protected subject matter for Functionality SRs should not be limited to functionalities indubitably amenable to protection through today's software patents. For example, a Functionality SR – ultimately replacing today's patent protection – should also be able to protect **complex simulations** or functionalities of **neural networks**, even where they may not be protectable under the COMVIK approach.

240 A Functionality SR would protect a specific solution to a specific problem. The subject matter of such a right would be neither an abstract goal nor an unlimited number of potential solutions to a problem. Instead, the applicant would have to claim **one or several concrete ways ("modalities") in which a computer program**, running on a machine, **performs the functionality**. The protection would only extend to these modalities.[308]

241 Specific **software architecture** can be key to the performance of a functionality. A Functionality SR could protect software architectures provided they fulfil, or form part of a claimed subject matter that fulfils, the protection requirements.

242 While modalities can be abstractly defined in the application (as in patent law's common practice), the **specifications must contain code** which demonstrates that and shows how the claimed modalities can be implemented. This excludes claiming abstract functionalities and ensures that a Functionality SR is only granted if the applicant demonstrates that the claimed **modality can be performed** by the source code disclosed in the application.

243 Although it seems possible that a single modality will obtain stand-alone protection, Functionality SRs will likely most often claim a **combination of modalities**. Frequently, most of these

---

306 See e.g. Art. 53(a) EPC: "contrary to 'ordre public' or morality".

307 This IP right provides protection for the underlying function of a software since this function determines the value of the product. Protection is not limited to the mere manifestation of the concept in a particular computer code expression; see WEINZINGER/SONN, GRUR Int. 1995, 747.

308 Such an approach would likely engender a narrow concept of the doctrine of equivalents, meaning that alternative ways to perform a functionality should not easily qualify as infringing equivalents.

modalities will already be known individually, but now for the first time they are being combined to perform a specific task in a specific way. Protection would then **extend only to the specific combination of modalities** that is able to solve a specific problem in a novel way. For **truly novel single modalities**, the situation is similar to groundbreaking patents which, though rare, tend to provide a broader scope of protection.

244 To **illustrate** the above using the **example** of **pedestrian simulations**: a Functionality SR would not permit to claim exclusivity for any software (possibly even software developed in the future) with the purpose of simulating optimized flow of people. Rather, protection could only be claimed for a specified software architecture that optimizes people flow in a specified manner based on a specified set of parameters (e.g. width of passage, average walking speed, typical reactions to signs or colours).

245 The patent on the **page rank algorithm**[309] used by Google to sort search results provides **another example** for how Functionality SR claims should be formulated. It concerns a computer-implemented method to enhance the performance of linked databases (i.e. search engine results).[310] To measure the importance of a search result, the algorithm counts the number and the quality of the links that lead to a document (e.g. a webpage) and ranks it accordingly within the search results.[311] It would be too unspecific to claim an algorithm that scores and ranks webpages. The modality claim(s) would have to exactly describe the factors which will influence the weight of each link (i.e. the number of the links to one page),[312] the probability that a user would access the link,[313] the visibility and or textual emphasis of a link on a webpage,[314] user-specific preferences,[315] etc. Furthermore, it would have to set out how the respective results are achieved and how the different methods work together to arrive at the overarching ranking method.[316] Furthermore, the specifications would have to contain code which demonstrates that and how the claimed modality(ies) can be implemented.

246 Subject to further discussion, functionality protection may also have a **sectoral limitation of scope**, i.e. a Functionality SR may be able to claim a modality only for a specific sector or context instead of across all potential use contexts.

247 To look at the issue from another perspective, a protectable modality is **not tantamount to its underlying idea/concept or to an algorithm/instruction to the human mind** it embodies. The idea that a device can be unlocked by sliding a finger over the screen (slide to unlock) or the idea that a document can be printed electronically in a file that cannot be altered (pdf) would not constitute a modality. The function that can be the subject matter of a Functionality SR is the actual **code implementation of the abstract idea or concept in a series of steps that can be performed on a machine**.

248 While distinguishing between the abstract idea/concept/algorithm/instruction to the human mind and the implementation of a functionality in a series of concrete steps may seem relatively

309 US 6285999.

310 US 6285999, 1.

311 US 6285999, 6 et seq.

312 US 6285999, 10 Claim.

313 US 6285999, 10 Claim 3.

314 US 6285999, 10 Claim 6.

315 US 6285999, 10 Claim 7.

316 US 6285999, 7 et seqq.

straightforward from a theoretical perspective, drawing a line in specific cases will be difficult at times. The adequate threshold will have to be **identified on a case-by-case basis** by IP offices and courts when deciding on a specific application or validity challenge. As with all other complex questions of IP law, this approach should **in due course** offer a **fair amount of legal certainty**, while allowing for the necessary **flexibility** to decide complex cases.

249 As for the Code SR (see above, F.III.2.a)), appropriate **exclusions** should prevent the granting of Functionality SRs for functionalities that **violate fundamental societal interests**.[317]

3. Requirement for Protection

a) Code-Level Software Right

250 The writing of source code is time consuming and requires programming skills and sector-specific knowledge. It would therefore be conceivable to protect source code regardless of whether it embodies a certain level of creativity or innovation. In this respect, a Code SR would be similar to the existing neighbouring rights in copyright law that protect subject matter such as performances, recordings and photographs without the need to fulfil any protection requirement. Alternatively, protection by a Code SR could require a **certain level of creativity or innovation** reflected in an appropriate, software-specific requirement for protection. This might be important for shorter programs and for many AI-generated programs which would be protected even where a relevant investment in respect of time and skills is lacking.

251 Defining an **appropriate protection requirement** for source code is not straightforward. It should reflect the main objective of the Code SR, which is to encourage the development of novel source code. While it is clear that simple replication of (parts of) existing code and minor adaptations would not be sufficient to grant protection, a **strict and objective novelty of the code cannot be required**, as it is impossible to verify novelty in the absence of a register or database containing (virtually) all existing source code.[318] The protection requirement should also reflect that the source code is written to implement a specific functionality(-ties) in a specific programming language, taking into account good programming practices and established programming standards. Given these restrictions, the **scope for creative and innovative solutions is often limited**, and in many cases different human programmers or AI systems would achieve a similar result.

252 Against this background, source code should only be protected by a Code SR if the software developing entity (e.g. companies, AI) or individual programmer has **used its leeway, which is potentially limited but still available, to take creative and/or innovative code-writing decisions** in such a way that the **probability of an independent creation of (virtually) identical source code appears to be very small**. As stated above, the assessment of the requirement of protection should be made from the **perspective of an expert,** taking into account **standard coding practice** (see, above, F.II.1.b))**.** The requirement for protection will often (but not necessarily always) be met by longer programs that are not created by simply compiling existing program parts. Code that was generated by an AI system upon basic prompts will usually not meet this requirement. This may be different where an AI system was specifically trained to generate specific, sophisticated types of computer programs (e.g. image compression).

---

[317] See e.g. Art. 53(a) EPC: "contrary to 'ordre public' or morality".

[318] Note that our approach would also encourage registration of Code SRs in the SR Ledger, but it would not make it a strict protection requirement (i.e. it provides for unregistered Code SRs).

253 As the acquisition of a Code SR does **not require pre-examination by an IP authority**, there is no presumption (even for registered Code SRs) that protection requirements are fulfilled. Instead, the **burden of proof lies with the alleged rights holder**, especially in case of a validity challenge.

b) Functionality-Level Software Right

254 It is sensible for all IP rights not to protect objects that already exist. This applies in particular to those IP rights that are primarily intended to promote innovation in a specific area, as is the case for the Functionality SR. As for other registered IP rights, it makes sense to require **novelty** of the claimed modality as the first precondition for protection. Novelty is assessed against the **state of the art at the time of application** for the Functionality SR, taking into account all existing software functionalities, including modality-implementing software not protected by a Functionality SR. This ensures that known modalities cannot be appropriated upon the introduction of the new Functionality SR.

255 **Code-level comparison** might be a helpful additional and/or initial parameter for novelty determination; if the claimed source code is too close to existing source code, this would tend to indicate lack of novelty.

256 Novelty alone, however, is not sufficient. The Functionality SR is intended to promote the development of innovative software. This would require an innovation step of sufficient quality beyond the state of the art, formulated for instance as a **non-obviousness requirement akin to patent law**. Such a requirement also forces an applicant to sufficiently narrow down the claimed modality so as to escape modalities already known. The relatively high threshold inherent in a non-obviousness requirement should **strongly reduce potential negative impacts of the Functionality SR on innovation**. This is all the more so if IP offices and courts endorse a strict interpretation of this requirement, in the basis of experiences made in patent law.

257 In addition to novelty and non-obviousness, a Functionality SR should only be granted if the applicant can **show that the modality can be executed on a machine**, namely by implementing it in source code.

4. Acquisition of Rights, Ownership, Transferability

a) Acquisition

aa) Code-Level Software Right

258 The Code SR should arise with the creation of the respective code. The protection should **take effect as soon as the code fulfils the protection requirements**. Prior examination of whether the protection requirements are fulfilled is not necessary, nor are any formalities.

259 It should be **possible though not mandatory to register** the source code in the SR Ledger. Such a registration would give the applicant a **demonstrable title** that will be helpful (e.g. for the enforcement of IP rights, M&A transactions, start-up financing and licensing). The registration could also establish a **presumption of ownership** of the registered Code SR holder. Finally, we suggest making an **extension of the initial term of protection** dependent on the Code SR's registration (F.III.5.b)).

260 Rights holders may be given the **option** to commission and pay for an examination of the protection requirements, at least in the form of an automated indicative **"machine examination"**,

in the event they register their Code SR. The IP title would have to **indicate whether or not the Right was registered with examination**.

bb) Functionality-Level Software Right

261 Functionality SR arise through their **entry in the SR Ledger**. The **initial, formal owner** of the right is the natural or legal person having applied for and been awarded the right.

262 Acquisition and registration of the Functionality SR would require an application. The **application** must **define the claimed modality(-ies)** in an abstract manner, similar to today's patent claims in a patent application. In addition, a **description** would be required (e.g. in the form of flow charts or other drawings). The application would have to also **explain how software is employed** to perform the claimed modality and contain source **code** which demonstrates that and how the submitted subject matter implements the functionality.

263 As a matter of principle, the Functionality SR could be granted with or without examination of the requirements for protection; it is also conceivable that only certain, but not all, requirements would be examined. As the Functionality SR can severely restrict third parties' freedom to operate, we believe that **protection should not be granted without a substantial preliminary examination**. It would seem sensible to carry out a complete preliminary examination that includes both novelty and non-obviousness. However, other solutions are conceivable as well.

264 As a **sidenote**, **AI systems** seem increasingly **able to easily support IP offices** in such an examination process by carrying out word- or code-based comparisons between new claims and existing SR Ledger entries or software registries.

b) Ownership

265 Persons who **developed modalities or wrote source code, or companies that organised and financed such development** (e.g. by using an AI system), can initially acquire SRs (i.e. corporate SRs are possible). Identity of who owns the right to the SR generated by a person at the behest of a company should depend on the **contractual arrangement** between these parties. Absent a clear stipulation, the company should acquire the right.

266 Where an **AI system generates code and/or software functionalities, the user of the system** should, as a default rule, **acquire Software Rights** to such output. The initial acquisition of such rights by the user of an AI system would not exclude **contractual arrangements** on the transfer of these rights to a third party (e.g. the provider of the AI system), if the arrangement is acceptable under general contractual rules including the control of general terms and conditions. The **user may also be the developer** of the system or be otherwise involved (e.g. providing training data), but this is not a requirement for the default rule to apply.

267 An **exceptions rule could add flexibility** to the above default rule, but this should be further assessed by research and expert discussion. **Parameters for allocating** Software Rights in exceptional cases could be: operating and financing the AI system; having developed the system; having trained the system; having contributed substantial training data willingly or unwillingly. **Bona fide rules** would apply and, in case of an SR violation between parameter-fulfilling persons, compensation claims would usually take precedence over injunctions.

c) Transferability, Licensing, and the SR Ledger

268 Both SRs would be **fully transferable and licensable** to other legal or natural persons. This also means that there would be no non-transferable moral rights (see below, F.III.5.a)).

269 In transactions over Software Rights, the SR Ledger would play an important role. **Transfers** of and **exclusive licences** to Functionality SRs would have to be **registered,** as this would cater to the need for transparency and reliable transactability described above.

270 The Ledger **should exercise strong protection** for bona fide **acquirers/licensees** of registered SR (including in cases of succession in ownership).

271 For appropriate transactions, the SR Ledger should **enable "smart" (licence) contracts**, i.e. automated, software-based contracting.[319] An exemplary setting could be the taking of standard (open source) licences for code listed in a library and which a programmer intends to use for a new computer program.

272 The authority administrating the SR Ledger would have to verify that the **Ledger generally provides** this feature. SR holders could add a **smart contracting option** (possibly including a pertinent API) to their registrations if they wish to enable an automated granting of licences for their software. In case of mandatory limitations, such an option may also be mandatory where the limitations' characteristics permit it (e.g. limited complexity).

273 The SR Ledger would likely have further beneficial effects. Where the claim/subject matter of a Functionality SR consists of known elements and distinctive novel elements, which are key to its protectability, referencing to pertinent Ledger entries could **remove the need to describe the known elements in detail**.

5. Effects of Protection

a) Exclusive Rights

274 The SR should cover **any use of the protected subject matter**. This would include the use of the source code or the protected modalities in the form of a computer program on a machine. Furthermore, the SR would cover the **reproduction and distribution** (including licensing) of copies of such a program on electronic and other data carriers, and the making available of such copies via networks, especially the internet.

275 The SR would cover **both commercial and private use** of the protected subject matter, but it would **not provide for** specific **moral rights**.

276 The scope of protection of the Code SR would encompass **all uses of identical and sufficiently similar code**, including translation into a different programming language. Although in principle the scope of protection of the Functionality SR should be narrow in order to avoid negative impacts on innovation, it should not only cover the implementation of the modality in any source code in the exact same way as claimed in the registration, but should also encompass **obvious equivalences**, lest the protection can be circumvented too easily.

---

[319] On smart contracts in general, see FURRER, Anwaltsrevue 2018, 103. On their use in a register context, see MEITINGER, Informatik Spektrum 2019, 371. On their use in the context of IP, see SCHAWE, MMR 2019, 218.

a) Limitations

277 Functionality-based protection especially can lead to relatively far-reaching **restrictions of market or innovation activities by third parties**. The **Functionality SR** would therefore have to be restricted by appropriate **limitations**.

278 Although the risk of impeding innovation is smaller with the Code SR, **limitations are equally important to ensure that the Code SR does not restrict future code development**, **interoperability** with other programs, and a **seamless and secure use** of the program. While many limitations would be more relevant for the Code SR or the Functionality SR, only certain limitations will be clearly irrelevant for one of the rights (on dependent programs and lack of use see below). We suggest a generally uniform regime of limitations for both types of Software Rights.

279 The SR should be restricted by at least the following limitations:

- **Research privilege**: As the SR should not hinder further development of computer programs, a comprehensive research privilege should be provided to ensure that the protected source code and functionalities can be used and further developed by third parties as a subject of research, free of charge and without further procedural requirements. The research privilege should protect both commercial and non-commercial uses, along the lines of current patent and copyright law.
- **Reverse engineering/decompilation**: Decoding of a computer program should be allowed for interoperability purposes. While this limitation is more important for the Code SR, it may also be relevant for the Functionality SR (see above, F.II.2.c) and F.II.1.d)bb)).
- **Bug fixing (incl. for security purposes):** Use of a computer program should be allowed if necessary to fix a bug or to ensure IT security. As with the limitation for reverse engineering/decompilation, the limitation for bug fixing is more important for the Code SR but may also be relevant for the Functionality SR (see above, F.II.2.c) and F.II.1.d)bb)).
- **Backup copy**: Anyone who rightfully uses a computer program should be allowed to make a backup copy of it. Even though such copies are less relevant in an environment ever more dominated by cloud computing and software as a service (SaaS), backup copies may be important to ensure resilience, (e.g. in case of a serious cybersecurity event).
- **Dependent programs**: A conspicuous case for a limitation would be "dependent programs". Computer programs with novel functionalities often use existing functionalities that may be protected by a Functionality SR. If such a computer program fulfils the requirements for protection of the Functionality SR (i.e. it is novel and non-obvious), the owner of the latter right should be able to request from the owner of the former right a licence which allows for the use of the protected functionalities. This mechanism enables further development of software functionalities by building on existing ones, and ensures that the owner of the pre-existing Functionality SR can claim fair remuneration for the use of the protected functionalities. This limitation should avoid unwanted effects a Functionality SR might have on innovation in the software industry if there were no limitation for the use of existing functionalities in dependent programs. It is tailored to the Functionality SR and does not apply to the Code SR.
- **Lack of use**: Another potential case for a limitation is the lack of use of a Functionality SR by the rights holder within a certain period of time after acquisition of the SR. A right to request the granting of a licence, based on lack of use, could emerge with the first extension of the Functionality SR after the initial 5-year protection period. This limitation should avoid the blocking of functionalities by the owner of a Functionality SR. Given that there are no similar risks on the code level, this limitation only applies to the Functionality SR.

- **Training, validating and testing of AI systems** (see above, F.II.1.d)bb).

280 While these may be the most important cases for limitations, there **may be other instances** mandating such limitations, including limitations to protect competition and dynamic efficiency. Furthermore, the limitations suggested heretofore require **refinement**, including their tailoring to specific settings. With regard to the limitations' aspect of our Novel Approach, we invite **further research and discussion**.

b) Term of Protection

281 The minimum term of protection should **correspond** roughly to the **average time** companies need to **amortize** their investment in developing the modalities and/or writing the source code, including the time needed to make a reasonable profit. Considering the rapidity of software innovation and development, the average recoupment time effectively available to developers is short. There are programs or parts of programs, such as the kernels of successful operating systems, whose use and commercial value extend over a long period of time. But this is the exception, and the long-term use of a program does not necessarily indicate the need for an equally long recoupment period, let alone the need to grant a longer protection in order to provide sufficient incentives to invest in developing the program's modalities and writing the source code.

282 With this background as a starting point for further discussion,[320] we propose an initial protection period of five years.

283 This initial period is **renewable**, subject to the payment of a **renewal fee**. These fees should be structured **progressively**, as is currently the case in patent law, to provide incentives to refrain from upholding an SR if the protected functionalities or code are not actually used by the owner of such right and/or its licensees. **Renewal of the Code SR should be subject to the registration of the Code SR**, but not to making the source code available on the SR Ledger. This is because rights holders must, in principle, remain free to decide whether and to what extent they wish to make the source code available to third parties.

284 The **maximum term of protection** after renewal should be no longer than **15 years**, i.e. an initial five-year term of protection which could be renewed twice.

285 When considering this proposal, it should also be noted that **computer programs are usually updated, adapted and optimized during their lifetime**. For such revised/added parts of the source code, the **term of protection begins anew**, provided the parts fulfil the protection requirements.

286 If research **should show that a longer period of unrestricted protection is mandated**, it is conceivable to react not only by adjusting the overall term of protection, but also by **a differentiated limitations regime**. For instance, compulsory licenses could take effect only after the initial five-year protection period.

[320] Determining an appropriate term of protection for the SR requires an in-depth economic analysis taking into account a series of factors, not least the average recoupment time in future software markets that will rely heavily on the use of AI. Such an analysis cannot be carried out in this study and should be the subject of future research.

6. The Software Right at a Glance

Novel Approach: sui generis Software Rights at a glance		
	Code SR	**Functionality SR**
Protected subject matter	Source code in all its forms	Modality: specific performance of a functionality by a computer program or software architecture running on a machine; possibly sector-specific
Protection requirement	Creative/innovative coding decision, considering limited leeway for creativity	Novelty and non-obviousness, successful performance on a machine
Acquisition	• Initial acquisition ipso jure upon code generation; no examination or formality requirement • (AI-based) examination and registration optional, registration benefits (see below)	• Application and SR Ledger registration • Full-fledged examination
Ownership	• Starting point: coder or modality developer • Corporate acquisition and ownership possible • AI systems: user default owner of output, contractual arrangements possible • Possibly exceptions (assessment of entitlement parameters) for special settings	
Effects of protection	• Any commercial or private use of protected subject matter, including reproduction and distribution • No moral rights • Identical and sufficiently similar code in scope	• Modalities as claimed, in any source code expression running on a machine • No moral rights • Obvious equivalences in scope
Transferability, licensing, SR Ledger	• Fully transferable, licensable; no non-transferable moral rights • Transfers of registered Software Rights under registration requirement • SR Ledger with strong (bona fide) protection for acquirers/licensees (including in cases of succession of ownership) • SR Ledger to provide smart contracting feature, e.g. for automated standard licences	
Minimum limitations (further differentiation between SR types subject to discourse)	• Protectability exclusion based on fundamental societal interests • Research privilege • Reverse engineering/decompilation for interoperability purposes • Bug fixing and IT security • Backup copy • Dependent programs • Lack of use • Training of AI systems, if not otherwise secured; possibly to be compensated	
Term of protection (subject to discourse)	• 5 years initially • Extension possible; extension fee (increasing); registration requirement for Code SR extension • 15 year maximum protection period	

7. Compatibility with International Law and Implementation Perspective

287 We believe that the Novel Approach described heretofore **complies with international law**, such as TRIPS, the Berne Convention, EPC, PCT, WCT and other treaties. These international agreements **do not restrict the introduction of new IP rights**, such as the SR, since they only address and harmonise the IP rights existing at the time they took effect.

288 Briefly, the aforementioned international treaties seek to harmonize a minimum protection level for subject matter deserving protection under IP rights addressed by the treaties. The optional availability of an additional IP right, and several characteristics of the SR (e.g. no human developer requirement, corporate rightsholdership, transactional benefits from (optional) registration) amount to an **overall *extension* of available IPR protection for computer programs not prohibited by minimum protection requirements**.

289 Furthermore, most elements of the Novel Approach would be **in line with international law if implemented as improvements to the current patent/copyright law framework** (see above, E.II), and if presented as a fallback, piecemeal solution. This shows that the characteristics of the Software Right would not violate the essence of the three-step test, to the extent it is perceived to be an overarching principle and requirement in IP law.

290 The best way of implementing the Novel Approach would be to **replace the current protection** granted by patent and copyright law with the new SR. However, **Art. 10(1) TRIPS**, which obliges all WTO member states to protect computer programs (source and object code) as literary works in accordance with the Berne Convention, does not allow for such a step. As long as this provision remains unchanged, an SR would have to **coexist at least with the protection of computer programs provided by copyright laws**. This raises important concerns as to **potential contradictions between the new SR and traditional copyright law**. For example, limiting the term of protection of the SR to 15 years seems to make little sense if copyright protection is granted for 50 years after the death of the author. However, we believe that an appropriate interpretation of patent and copyright law could provide **strong incentives** to **rely on Software Rights** instead of patent and copyright law.

291 Given the availability of a tailored Functionality SR, the **EPO and national courts could adapt their interpretation of the EPC** and pull back on granting patents for CII, based on a restrictive reading of the concept that no patents are granted for software "as such".

292 A similar effect could be achieved in copyright law. At least in Continental European copyright laws, copyright protection is only granted to intellectual creations, i.e. literary and artistic works (or computer programs) created by human beings (see above, D.IV.1.b)aa) and E.VI). **Given the availability of an SR which would not require a human creation, national courts could restrictively interpret the traditional requirement of human creation** by only granting copyright protection to those computer programs or parts thereof which are actually, clearly and fully created by human beings. A strict application of this requirement would force rights holders to substantiate and prove in court proceedings which parts of the source code, if any, were written by a human being. In most cases this would be impossible or simply too burdensome.

293 If factually and over time SRs replaced the use of software patents/copyrights due to market preferences, such a development would not violate international treaties as they do not impose IPR usage obligations on market participants. **If at some point national legislation or an international agreement envisaged replacing software-patent/copyright protection by Software Right protection, the compatibility of such a step with international law**, including amendments

to international treaties necessary in the course of it, **could be assessed anew**. There is no need to cross this bridge today.

G. Impact on Future Tasks of IP Offices, Especially the IPI

294 This study has shown a need for action to adapt the current IP protection for computer programs in patent and copyright law, in order to ensure adequate protection for software that is fully or partially generated by AI systems and to support companies with software-related business models. The findings of this study suggest **four areas of possible engagement** for IP offices. Two of them concern action which the IPI could take. The other two are the authors' aspirations for a more distant future.

I. Potential Next Steps

- **Further research:** Acknowledging that this study is only a first step towards a tailor-made IP protection regime for computer programs, further research is essential. IP offices (including the IPI) should promote and participate, mainly as a dialogue partner, in further research. Among the important topics for further in-depth research are (i) the amendment and adaptation of today's **software-specific limitations** in copyright law; (ii) the possible **development of European patent law** towards a functionality-based granting of software-related rights; (iii) features and design of a ledger-based, smart, contract-ready **software register**. A potential way forward would be to enable and participate in international expert and stakeholder workshops focusing on software protection.
- **Scientific and (national) policy dialogue:** Based on the present study and further research, IP offices (including the IPI) should support a dialogue between national and international experts, stakeholders (companies, counsel, open-source community, etc.), and policy leaders. Such a dialogue could also evaluate the options for legislative action in Switzerland. An international workshop, co-hosted by the IPI and open to national lawmakers, could initiate such dialogue.

II. Future Perspectives

- **International Working Level Dialogue**: IP office representatives at the working level could engage in exchange and cooperation on various elements of the improvements highlighted in this study. A conspicuous example is the design, building and operating of a **software register** ("**SR Ledger**"), or the update of IP examination guidelines.
- **International Policy Dialogue:** Ideally, IP offices should work towards an international policy dialogue on the problems and potential solutions identified in this study. This dialogue should involve IP offices from selected countries around the world, as well as WIPO and possibly the WTO. As a first step, IP offices should work towards establishing a **common understanding of the challenges for software protection** – both long-standing issues and novel challenges caused by software-generating AI systems – as well as understanding of the room for manoeuvre within the current system of international treaties, in view of differing legal frameworks/traditions. In the long run, IP offices, WIPO and the WTO should also explore ways forward to adapt the TRIPS Agreement and/or the Berne Convention, or to create a **new international treaty specifically for software**. We believe that **in the long run** the **IPI could be in a position to become a highly visible, driving force** in such a dialogue, not least because it operates in a leading

digital economy[321] and has already built more knowledge and a stronger network regarding this topic than many other IP offices.[322]

[321] See the large-scale operations of companies like Google, IBM and Microsoft in Switzerland, as well as the country's blockchain and crypto industry.

[322] See the work done and networks established by the IPI in the course of its collaboration with University of Zurich's CIPCO in the field of AI and IP.