

Rechtshandbuch Legal Tech

Breidenbach / Glatz

2. Auflage 2021
ISBN 978-3-406-73830-2
C.H.BECK

ohne dass es der Mitwirkung der Parteien beziehungsweise des Rückgriffs auf klassische Durchsetzungsmechanismen einer nationalen Rechtsordnung bedürfte.

Damit ein digitaler Vertrag automatisiert abgewickelt werden kann, muss die Laufzeitumgebung, in der dieser ausgeführt wird, Schnittstellen zu den wirtschaftlichen Vermögenswerten bereitstellen, welche im Laufe der Vertragsabwicklung ausgetauscht werden sollen. An diesen neuralgischen Punkt knüpfen neue Entwicklungen, wie digitale Eigentumsrechte, digitale Währungen und das Internet of Things, an.

B. Hintergrund: Ideengeschichte softwarebasierter Verträge

I. Computable Contracts: Rationalisierung durch Formalisierung

Typischerweise erfolgt die Niederschrift eines geschlossenen Vertrages zu Beweis Zwecken in natürlicher Sprache, beispielsweise auf Deutsch. Doch selbst, wenn sprachlich und strukturell hoch formal, bleibt auch ein in juristischer Fachsprache verfasster Vertragstext für Computer eine weitgehend unverständliche Aneinanderreihung willkürlicher Zeichenketten.⁶ Dabei verspricht die Digitalisierung von Verträgen großes Potenzial für die Senkung von Transaktionskosten, die Schaffung von Rechtsklarheit und die Minimierung von Vertragsverstößen. Digitalisierung in einer Weise, welche die Vertragsversprechen der Parteien für einen Computer zumindest insoweit verständlich machen, als dass ersichtlich wird, welche konkrete Leistung von welcher Partei unter welchen Bedingungen versprochen wurde.

Ein Vertrag kann unabhängig von dem Medium, in welchem die Parteien dieses zu Beweis Zwecken perpetuieren, seien es Papier und Feder oder digitale Bits und Bytes, als ein *formalisiertes* Leistungsprogramm begriffen werden, auf das sich die Parteien geeinigt haben. Aufbauend auf diesem Gedanken ist die formale Spezifizierung von Verträgen in *dry code*⁷ und die damit verbundene computergestützte Analyse (sog. *Computable Contracts*) schon seit geraumer Zeit Gegenstand der Forschung in der Informatik.⁸ Praktische Anwendung fand diese Forschung bislang im Bereich des *Contract Lifecycle Management* (CLM) sowie dem *Financial*

⁶ Die parallel zu den Smart Contracts evolvierende Technik des *Machine Learning* benutzt rein statistische Methoden um die Illusion echten Verstehens zu erwecken, vgl. → Kap. 3.1. Rn. 20 von *Bünau*, Künstliche Intelligenz – Möglichkeiten und Mythos.

⁷ Natürliche Sprache ist „wet code“: nur das menschliche Gehirn – die „wet ware“ kann sie vollständig begreifen. Sie ist oft mehrdeutig und lebt vom Kontext. Computercode hingegen ist klar, präzise und eindeutig. Code ist daher „dry code“: er wird ausgeführt auf der „trockenen“ Hardware eines Computers. Zur Prägung des Ausdrucks „wet code“ und seinem Gegenteil „dry code“, vgl. *Szabo*, Wet code and dry, 2008, <http://unenumerated.blogspot.com/2006/11/wet-code-and-dry.html>, zuletzt abgerufen am 1.3.2021. Auch → Kap. 9.2. Rn. 6 ff. von *Breidenbach*, Rulemapping.

⁸ Vgl. *Surden*, Computable Contracts, UC Davis Law Review, Vol. 46, 2012, pp. 629–700 und *Flood/Goodenough*, Contract as Automaton: The Computational Representation of Financial Agreements, 2015, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2613869, zuletzt abgerufen am 1.3.2021.

*Engineering*⁹. Erklärtes Ziel ist es, einen Formalismus zu finden, der sowohl in der Lage ist eine Vielzahl der Charakteristika eines typischen vertraglichen Leistungsprogramms auszudrücken, als auch „ausführbar“ ist, das heißt als ein System aus Zuständen und Übergängen modelliert werden kann, welches die reale Vertragsausführung abbildet.

- 10 Einen ähnlichen, wenn auch viel bescheideneren Ansatz, Verträge als formale Systeme zu begreifen, sind kommerzielle Dienste wie smartlaw.de, welche die Erstellung personalisierter Vertragsdokumente per Mausclick anbieten.¹⁰ Auch hier antizipiert ein formales Modell die entscheidenden Parameter eines bestimmten Vertragstyps und lenkt auf dieser Basis Rechtslaien durch einen Fragenkatalog, an dessen Ende ein strukturierter Vertrag steht, der zwar in natürlicher Sprache formuliert ist, dessen Komposition aber mit einem formalen Modell dieses Vertrages und seinem antizipierten Lebenszyklus korrespondiert.
- 11 Aus der Betrachtung von Verträgen als formale Modelle ergeben sich signifikante Rationalisierungseffekte. Formale Systeme, wie sie etwa in der Automaten- und Modellierungstheorie zur Modellierung zustandsbasierter Systeme herangezogen werden, zwingen den Vertragsverfasser, komplexe Lebensumstände als strukturierten Geschehensverlauf in Form diskreter Zustände auszudrücken. Es ist also ein mentaler Übersetzungsakt gefordert, der für sich genommen schon zu einem kohärenteren Vertragswerk führen kann. Ebenso wird die Möglichkeit eröffnet, Verträge in standardisierte und interoperable Module zu unterteilen. Sodann kann die Anwendung maschinengesteuerter Optimierungsprozesse die Qualität des Vertrages weiter verbessern, indem von einfachen numerischen Rechenfehlern bis hin zu komplexen Compliance Fragen eine Vielzahl von Problemen aufgedeckt werden können.¹¹

II. Smart Contracts: Automatisierung durch Standardisierung

- 12 Einen bedeutenden Schritt weiter geht die Idee der *Smart Contracts*. Wie auch ein *Computable Contract*, zeichnet sich ein *Smart Contract* zunächst dadurch aus, dass er das vertragliche Leistungsprogramm in maschinenlesbarer Weise nachzeichnet und einen Computer damit in die Lage versetzt zu verstehen, wer wem was unter welcher Bedingung schuldet. Im Gegensatz zum Konzept eines *Computable Contract* dient ein digitaler Vertrag aber nicht lediglich der Modellierung und Planung betrieblicher Ressourcen innerhalb eines Contract Lifecycle Management Systems, sondern übernimmt vielmehr die *tatsächliche* Vertragsausführung. Ein Smart Contract ist deshalb in einem faktischen Sinne verbindlich, denn er vermittelt den realen Leistungsaustausch zwischen den Vertragsparteien.¹²

⁹ Vgl. Jones/Eber/Seward, Composing contracts: an adventure in financial engineering (functional pearl), in Proceedings of the 5th ACM SIGPLAN international conference on Functional programming, ACM, 2000, pp. 280–292.

¹⁰ Vgl. → Kap.5.5. Rn.12 von Wend/Gebhardt, Vertragsgeneratoren und Vertragsmanagement.

¹¹ Vgl. → Kap.9.1. Rn.9ff. von Wong, Computable Contracts – from Academia to Industry.

¹² Nach Definition von Kaulartz/Heckmann ist ein Smart Contract „eine Software, die rechtlich relevante Handlungen (insbesondere einen tatsächlichen Leistungsaustausch)

Die ursprüngliche Idee der Smart Contracts ersann der amerikanische Jurist 13
 Nicolas J. Szabo („Nick“ Szabo) in den frühen neunziger Jahren, als das Internet
 erstmals flächendeckend jedem interessierten Laien zur Verfügung stand und der
 bislang unerforschte Möglichkeitsraum des „Cyberspace“ die Gemüter libertärer
 Denker erregte.¹³ Szabo erkannte, dass eine Welt, in der sehr viele Informationen
 digitalisiert sind und das Internet sowohl ein Ort signifikanten wirtschaftlichen
 Handels darstellt sowie viele andere Arten der privaten und geschäftlichen Kom-
 munikation vermittelt, der Bedarf für einen „nativ digitalen“ Vertrag besteht.
 Diese Art von Vertrag nannte er *Smart Contract*.¹⁴

Ein *Smart Contract* kann wegen seiner koordinativen Wirkung als das digitale 14
 Äquivalent zu einem klassischen Rechtsvertrag verstanden werden. Zwei oder
 mehr Parteien gehen Vertragsversprechen ein, welche sie im Nachgang des formalen
 Vertragsschlusses erfüllen. Der Unterschied eines *Smart Contract* zu dem
 Vertrag wie er einem klassisch geschulten Juristen vertraut ist, kommt im Falle der
 zwangsweisen Durchsetzung vertraglicher Ansprüche zum Tragen. Im Falle der
 Störung einer primären Leistungspflicht können die Gläubiger eines klassischen
 Rechtsvertrages die Erfüllung der geschuldeten Leistung durch Anrufung der
 Gerichte oder anderer Organe des Rechtswesens erzwingen beziehungsweise das
 enttäuschte Vertrauen ersetzt verlangen. Ein *Smart Contract* hingegen operiert
 in einem vernetzten digitalen Medium. Dadurch kann er über die vertraglich zur
 Dispositionen gestellten Vermögensgegenstände selbstständig verfügen und so die
 Vertragserfüllung automatisieren und erzwingen. Der Vertrag wird dadurch zum
 Treuhänder. Erzielt wird dieser Effekt durch neue Entwicklungen im Bereich des
Trusted Computing, also der vertrauenswürdigen Ausführung von Software.¹⁵

Wofür *Smart Contracts* heute tatsächlich schon eingesetzt werden können, ist 15
 letztlich die Frage danach, was schon über eine offene digitale Schnittstelle zum
 Internet verfügt. Denn ein *Smart Contract* hat potenziell Zugriff auf alle Ressourcen,
 die vermittelt durch das Internet über eine Schnittstelle ansprechbar sind.
 Aufgrund der stetig wachsenden Digitalisierung aller Bereiche des alltäglichen und
 gesellschaftlichen Lebens, vom *Smart Home*, zum *Smart Car*, zur *Smart City* und
Smart Money: Körperliche und nichtkörperliche Gegenstände und Systeme finden
 sich mit wachsender Geschwindigkeit als Steuerungsschnittstelle im globalen ver-

in Abhängigkeit von digital überprüfbar Ereignissen steuert, kontrolliert und/oder
 dokumentiert, mit dessen Hilfe aber unter Umständen auch dingliche und/oder schuld-
 rechtliche Verträge geschlossen werden können“, vgl. Braegelmann/Kaulartz, Einteilung,
 in Braegelmann/Kaulartz, Rechtshandbuch Smart Contracts, S. 8.

¹³ Barlow, A Declaration of the Independence of Cyberspace, 1996, <https://www.eff.org/de/cyberspace-independence>, zuletzt abgerufen am 1.3.2021.

¹⁴ Szabo, Formalizing and Securing Relationships on Public Networks, First Monday 1997, <http://firstmonday.org/ojs/index.php/fm/article/view/548/469>, zuletzt abgerufen am 1.3.2021.

¹⁵ Blockchains (vgl. → Kap. 4.1. Rn. 20ff. von Glatz, Blockchain – ein Paradigmenwechsel?) ermöglichen die vertrauenswürdige Ausführung von Software durch ein Computernetzwerk, indem sich die Netzwerkteilnehmer gegenseitig daraufhin kontrollieren, ob die Ausführung der Software tatsächlich regelkonform geschieht. Betrugsversuche werden erkannt und von der „ehrlichen Mehrheit“ ignoriert.

netzten Raum wieder. Dieser Raum ist es, welcher die Idee der *Smart Contracts* zu einem interessanten Phänomen für das digitale Zeitalter werden lässt.

- 16 Zusammenfassend zeichnet sich die Qualität eines Smart Contract Systems im Wesentlichen durch zwei Faktoren aus: der Fähigkeit, vertragliche Vereinbarungen durch formale Sprachen in maschinell verständlicher Weise nachzuzeichnen sowie der Verfügbarkeit digitaler Schnittstellen zur Ermöglichung softwaregesteuerter Transaktionen über bestehende Vermögensgüter.

C. Heutiger Entwicklungsstand der Smart Contracts

I. Zusammenhang zwischen Blockchain-Technologie und Smart Contracts

- 17 Die Erfindung der Blockchain-Technologie bedeutet einen Quantensprung hin zur praktischen Realisierbarkeit des Konzepts der Smart Contracts. Die Netzwerktechnologie liefert zwei bedeutende Elemente, die den Ideenvätern der nativ digitalen Verträge in den Neunzigerjahren noch fehlten. Nick Szabo, der Namensgeber der Smart Contracts und heißester Kandidat für die wahre Person hinter dem Pseudonym des bis heute unbekanntes Bitcoin-Erfinders *Satoshi Nakamoto*, definierte Smart Contracts zu Beginn der Neunzigerjahre folgendermaßen:

*“A smart contract is a set of promises, specified in digital form, including protocols within which the parties perform on these promises.”*¹⁶

- 18 Die Protokolle, auf deren Basis die Parteien eines Smart Contract ihre geschuldete Leistung erbringen sollten, waren zu der Zeit, in der Szabo diese Definition prägte, technisch und gesellschaftlich außerhalb jeder Reichweite. Es war schon rein konzeptionell nicht klar, worum es sich dabei handeln könnte. Sicher war nur, mit diesen Protokollen müssten *irgendwie* Leistungsversprechen „erfüllt“ werden können.
- 19 Die Blockchain brachte Licht ins Dunkel. Zum einen ist sie das Medium, in dem Smart Contracts ein autonomes Dasein führen können, die sogenannte *Contract Host Plattform*. Wenn ein Smart Contract erst einmal dort landet, verlieren die Parteien bis zu einem gewissen Grad die Kontrolle darüber. Der Vertrag wird zum Software-Agenten in der Laufzeitumgebung des Blockchain-Netzwerks und zum Treuhänder über die ihm anvertrauten Vermögensgüter. Er kann eigenständig auf Ereignisse reagieren und mit anderen Teilnehmern des Netzwerks in Austausch treten. Vor allem aber dient die Blockchain als neutrale Ausführungsumgebung für das vertragliche Leistungsprogramm.¹⁷ Das Netzwerk erzwingt die Einhaltung der definierten Regeln und lässt einseitige Änderungen oder Abweichungen nicht zu.

¹⁶ Szabo, Formalizing and Securing Relationships on Public Networks, First Monday 1997, <http://firstmonday.org/ojs/index.php/fm/article/view/548/469>, zuletzt abgerufen am 1.3.2021.

¹⁷ Vgl. *Matthes*, Smart Contracts: Eine Standortbestimmung aus Sicht der Informatik, in Braegelmann/Kaulartz, Rechtshandbuch Smart Contracts, S.44 ff.

Zum andern erfand die Blockchain eine vollkommen neue Form, Vermögensgüter digital zu repräsentieren. Zunächst in Gestalt der Bitcoins brachte die Blockchain-Technologie sogenannte Krypto-Assets hervor, die heute gemeinhin als *Token* bezeichnet werden.¹⁸ Ein Token kann alles repräsentieren, worauf sich die Parteien eines *Smart Contracts* geeinigt haben. Es handelt sich um eine standardisierte digitale Identität für einen Vermögenswert. Diese kann wiederum einem oder mehreren Eigentümern zugewiesen werden. So lässt sich mittels einer Blockchain jedes denkbare Wirtschaftsgut digital abbilden und über eine standardisierte Schnittstelle den *Smart Contracts* zur Disposition stellen.

II. Entwicklungsstand des Blockchain-Ökosystems

Trotz der rasanten technischen Fortschritte, die in den letzten Jahren erreicht worden sind, befindet sich das gesamte Ökosystem Blockchain noch in den Kinderschuhen. Es gibt gerade einmal eine einzige Blockchain, welche als *Contract Host Plattform* tauglich ist, namentlich Ethereum. Bitcoin, die nach wie vor größte Blockchain,¹⁹ beherrscht nur vergleichsweise primitive Möglichkeiten zur Automatisierung von Transaktionen. Alle anderen fortgeschrittenen Blockchain-Netzwerke sind entweder noch im Konzeptstadium, haben also noch kein großes, robustes Netzwerk aus Teilnehmern erreicht, oder weisen keine geeignete Ausführungsumgebung für *Smart Contracts* auf.

Die gängigen Programmiersprachen, in denen *Smart Contracts* heute programmiert werden, sind noch stark an den Spezifika der zugrundeliegenden Ausführungsumgebung verhaftet.²⁰ Sprachen mit höherem Abstraktionsniveau, die gegebenenfalls auf Lösungen in anderen Problemdomänen zugeschnitten sind, befinden sich noch im Konzeptstadium.²¹ Ebenfalls mangelt es weitestgehend an *best practices*, beziehungsweise werden diese nach diversen Sicherheitskandalen in der Community gerade entwickelt, um das allgemeine Sicherheitsniveau zu heben.²²

Ein Kritikpunkt, den *Smart Contracts* heute häufig ernten, wenn Anwälte anfangen sich damit zu befassen, ist, dass sie weder „*Contracts*“ im Rechtssinne seien noch in gesteigertem Maße „*smart*“. An der Oberfläche scheint diese Kritik wahr zu sein, doch verfehlt sie den Kern dessen, worum es eigentlich geht.

In der Tat sind *Smart Contracts* keine intelligenten Rechtsverträge. Sie scheren sich nicht um das Recht. Vielmehr machen sie das Recht überflüssig innerhalb ihres beschränkten Wirkungskreises. Ein *Smart Contract*, der auf einer öffentlichen Blockchain wie Ethereum ausgeführt wird, schafft eine Verbindlichkeit zwischen den involvierten Parteien, die nicht mal ein Rechtssystem zu garantieren vermag.

¹⁸ Vgl. → Kap. 4.1. Rn. 40ff. von Glatz, Blockchain – ein Paradigmenwechsel?; vgl. auch Sandner/Braunberger/Gabriel, Die Tokenisierung mittels *Smart Contracts* aus technischer Sicht, in Braegelmann/Kaulartz, Rechtshandbuch *Smart Contracts*, S. 29ff.

¹⁹ Gemessen an Metriken wie der Marktkapitalisierung der Währung sowie verfügbarer Rechenkraft im Netzwerk.

²⁰ Vgl. <http://solidity.readthedocs.io/en/latest/>, zuletzt abgerufen am 1.3.2021.

²¹ Vgl. → Kap. 9.1 Rn. 57ff. von Wong, *Computable Contracts – from Academia to Industry*.

²² Vgl. etwa das OpenZeppelin Projekt, welches ein offenes Repositorium an sicheren *Smart Contracts* entwickelt, <https://openzeppelin.org>, zuletzt abgerufen am 01.03.2021.

Denn man erfährt im Recht stets nur *ex-post*, ob ein Versprechen tatsächlich verbindlich und seine Einhaltung damit erzwingbar ist. In der Welt der Smart Contracts wird dieses Prinzip umgekehrt: hier ist schon *ex-ante* klar, dass die Parteien ihre Obligationen erfüllen werden.

- 25 Gerade wegen dieses Unterschieds sind Smart Contracts auch keine 1:1 Übersetzungen von juristischer Fachsprache in Software. Smart Contracts sind mit großer Sorgfalt entworfene Spielmechanismen, welche die Interessen einer Vielzahl nicht-kooperativer Spieler zu koordinieren und in ein Gleichgewicht zu bringen versuchen. Wer Smart Contracts für komplexe Transaktionen entwickelt, muss ein Hybrid aus Informatiker, Ökonom, Jurist und Spieltheoretiker sein.
- 26 Die Kritik, Smart Contracts seien abgesehen von ihrem mangelnden Vertragscharakter auch nicht „intelligent“, unterwirft den Kritiker demselben Verdacht. In der Regel wird hier bemängelt, dass den Smart Contracts die notwendigen Informationen beziehungsweise der Zugang zu den entsprechenden Schnittstellen fehle, um in einer Vielzahl von Situationen wahrlich autonom handeln zu können. Auch das ist oberflächlich betrachtet wahr. Doch führt gerade die *Digitalisierung* zur exponentiellen Zunahme digitaler Daten und Schnittstellen zur „analogen Welt“. Durch sogenannte Orakel-Dienste werden diese Daten über den *trust boundary* hinweg in das Innere der Blockchain-Netzwerke geleitet, wo sie dann den Smart Contracts als Automatisierungsgrundlage zur Verfügung stehen.²³

D. Smart Legal Contracts

- 27 Nachdem die Blockchain-Technologie einige Jahre ein Nischendasein führte, katapultierte die enorme Wertsteigerung mancher Kryptowährungen die Technologie in das Bewusstsein einer breiten Masse. Nunmehr ist auch eine Vielzahl von Juristen auf das Potenzial der Blockchain aufmerksam geworden und sie beginnen die Schnittstellen zum etablierten Vertragswesen zu erforschen und aktiv zu entwickeln.²⁴
- 28 *Smart Legal Contracts* beziehungsweise *Legal Smart Contracts* sind die gängigen Begrifflichkeiten, die zur Beschreibung dieser Entwicklungen verwendet werden. Im Kern wird hier versucht die Kluft zwischen der Welt der blockchain-basierten Smart Contracts und dem Rechtssystem zu überbrücken. Es handelt sich um ein Übersetzungsproblem im weiteren Sinne.²⁵ Die Motivation es zu lösen ist offensichtlich: Die grundsätzliche Unabhängigkeit der *Smart Contracts* von der Existenz des Rechtssystems bedeutet nicht, dass sie sich um dieses nicht

²³ ISDA, Smart Contracts and Distributed Ledger – A Legal Perspective, 2017, S. 18. http://www.linklaters.com/pdfs/mkt/london/Smart_Contracts_and_Distributed_Ledger_A_Legal_Perspective.pdf, zuletzt abgerufen am 1.3.2021.

²⁴ Vgl. → Kap. 9.1. Rn. 64 von Wong, Computable Contracts – from Academia to Industry, mit einer Auflistung von Projekten.

²⁵ Khalil/O, Brien/Butler, A Solution for the Problems of Translation and Transparency in Smart Contracts, 2017, S. 6, <https://e9a5d5c6.stackpathcdn.com/wp-content/uploads/2017/06/GR3C-Smart-Contracts-White-Paper-2017.pdf>, zuletzt abgerufen am 1.3.2021.

zu kümmern brauchen, wenn sie von einem Rand- zum Massenphänomen werden wollen.²⁶

Die Schnittstelle, an der das Rechts- mit dem Smart Contract System verschaltet werden soll, wird heute üblicherweise im Bereich der Sprache, den intelligenten Vertragsgeneratoren und dem Rulemapping²⁷ gesehen. Auf Ebene der Sprache werden domänenspezifische Sprachen (sogenannte DSL) entwickelt, welche es dem Vertragsverfasser erlauben sollen, in technisch wie rechtlich abgesicherter Form Verträge zu entwerfen, die einerseits soweit möglich auf einer Blockchain ausführbar sind, andererseits im Einklang stehen mit dem regulatorischen Umfeld des jeweiligen Vertrags.²⁸

Ebenso zählt die Isolation der Smart Contracts vom Datenreichtum des Internets und darüber hinaus zu einem der Kernprobleme, die für eine erfolgreiche Verschaltung der Rechts- und Smart Contract Systeme gelöst werden müssen.²⁹ Lösungsvorschläge gibt es bislang wenige, was daran liegt, dass das Problem im Grunde nicht lösbar ist. Daten über Ereignisse in der Welt (etwa Wetterdaten, Finanzdaten, Personendaten etc) einem Smart Contract verfügbar zu machen, ist im Kern ein Vertrauensproblem: Woher weiß der Vertrag, dass die Daten, die ihm gefüttert werden, authentisch sind? Kann er der Quelle vertrauen? Das Startup ChainLink versucht über ein dezentrales Orakel-Protokoll, Menschen überall auf der Welt dazu zu incentivieren, Datenquellen für die Blockchain bereitzustellen.³⁰

Zuletzt bedarf es auch Konfliktlösungsmechanismen für Smart Contracts. Nicht nur werden Konflikte zwischen den Parteien eines Vertrages auch in Zu-

²⁶ Zustimmend Möslein, der auch im Rahmen eines rechtlich nicht prädeteterminierten Begriffsverständnisses der Smart Contracts die Anwendung der Rechtsgeschäftslehre als maßgeblich betrachtet, vgl. *Möslein*, Rechtsgeschäftslehre und Smart Contracts, in Braegelmann/Kaulartz, Rechtshandbuch Smart Contracts, S. 82.

²⁷ Vgl. → Kap. 9.2. Rn. 3 ff. von *Breidenbach*, Rulemapping.

²⁸ Vgl. *Delmolino et al.*, Step by step towards creating a safe smart contract: Lessons and insights from a cryptocurrency lab, in *Financial cryptography and data security*, 2016, pp. 79–94; *Pettersson/Edström*, Safer smart contracts through type-driven development, PhD thesis, Master's thesis, Chalmers University of Technology & University of Gothenburg, Sweden, 2015; *Luu et al.*, Making smart contracts smarter, in *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, 2016, pp. 254–269; *Idelberger et al.*, Evaluation of logic-based smart contracts for blockchain systems, in *Rule technologies. research, tools, and applications: 10th international symposium, ruleML proceedings*, 2016, pp. 167–183; *García-Bañuelos*, Optimized Execution of Business Processes on Blockchain, ArXiv eprints 2016; *Frantz/Nowostawski*, From institutions to code: Towards automated generation of smart contracts, in *IEEE 1st international workshops on foundations and applications of self* systems (fAS*W)*, 2016, pp. 210–215; *Clack/Bakshi/Braine*, Smart Contract Templates: essential requirements and design options, 15.12.2016 ArXiv eprints 2016, pp. 1–15; *Diedrich*, Lexon, die Sprache der nächsten Revolution, *Rethinking Law 1/2018*, S. 18 ff.); *Genovese*, Smart Contracts and their Issues, *Rethinking Law 1/2018*, S. 28 ff.).

²⁹ Vgl. *Greenspan*, Why Many Smart Contract Use Cases Are Simply Impossible, 2016, <https://www.coindesk.com/three-smart-contract-misconceptions>, zuletzt abgerufen am 1.3.2021.

³⁰ *Ellis/Juels/Nazarov*, ChainLink, 2017, <https://link.smartcontract.com/whitepaper>, zuletzt abgerufen am 1.3.2021.

kunft unausweichlich sein und bedürfen deshalb der Möglichkeit, durch einen unabhängigen Dritten gelöst und durchgesetzt zu werden.³¹ Der Determinismus des *dry code* eines Smart Contracts sowie die Unabänderlichkeit der Verträge, sobald sie auf einer blockchainbasierten Contract Host Plattform installiert wurden, machen die Verfügbarkeit von derartigen Lösungen notwendig.³² Was sich hier durchsetzen wird, ist noch ungewiss. Das Kleros Projekt versucht etwa das Problem durch Crowdsourcing zu lösen.³³ Das Mattereum Projekt baut dagegen auf ein privates Schiedsgericht.³⁴

E. Ausblick

- 32 Werden Verträge in Zukunft durch Software abgebildet, wird die Ausgestaltung der Vertragsbeziehungen an die Fähigkeiten und Limitierungen von Software gekoppelt. Da das Verhältnis zwischen den Parteien im Zweifel aber auch ein rechtliches ist, entstehen Wechselwirkungen zwischen zwei zuvor weitgehend unzusammenhängenden Begriffs-, Verständnis- und Möglichkeitsräumen. Nur wenn die Interaktionen zwischen „Recht“ und „Software“ genau verstanden werden, kann die langfristige Entwicklung von Rechtssystemen in der digitalen Informationsgesellschaft nachvollzogen und bestenfalls aktiv mitgestaltet werden.
- 33 Vor diese Herausforderungen sehen sich Juristen und Informatiker heute gleichermaßen gestellt. Der größte gesellschaftliche Mehrwert entsteht dort, wo Juristen und Entwickler anfangen miteinander zu kommunizieren. Dass diese Kommunikation gelingt, ist mitunter Aufgabe von Legal Tech.

³¹ Zur Notwendigkeit im Rahmen insolvenzrechtlicher Streitigkeiten vgl. *Paulus/Braegelmann*, Smart Contracts im Krisenfall in Braegelmann/Kaulartz, Rechtshandbuch Smart Contracts, S.233.

³² Zustimmend *Paulus/Braegelmann*, Smart Contracts im Krisenfall, in Braegelmann/Kaulartz, Rechtshandbuch Smart Contracts, S.234.

³³ *Lesaege/Ast/George*, Kleros – Short Paper v.1.0.7, September 2019, vgl. <https://kleros.io/whitepaper.pdf>, zuletzt abgerufen am 1.3.2021.

³⁴ *Gupta*, Mattereum, <http://internetofagreements.com/assets/MattereumDraftforPublicComment.pdf>, zuletzt abgerufen am 1.3.2021.