Festschrift für Anton K. Schnyder

Herausgegeben von

Pascal Grolimund
Alfred Koller
Leander D. Loacker
Wolfgang Portmann

Schulthess §

Festschrift für Anton K. Schnyder

zum 65. Geburtstag

Herausgegeben von

Pascal Grolimund
Alfred Koller
Leander D. Loacker
Wolfgang Portmann

Schulthess § 2018

Bibliografische Information der Deutschen Nationalbibliothek
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über http://dnb.d-nb.de abrufbar.

ISBN 978-3-7255-7364-6

www.schulthess.com

Hans Rudolf Trüeb

Smart Contracts

The factory of the future will have only two employees, a man and a dog. The man will be there to feed the dog. The dog will be there to keep the man from touching the equipment. (Warren Bennis, 1991)

The contract of the future will have two parties, a machine and another machine. For some time, the machines will be operated by men. (Contemporary Chinese Wisdom, 2017)

Table of Contents Page

I. Introduction

The basic concepts of contract law have outlasted wars, technological change, the Roman Empire and even the new religion of state interventionism. The notions that agreements are binding *(pacta sunt servanda),* that default requires a reminder *(interpellatio),* or that nobody shall claim performance if he had not performed an agreement himself *(exceptio non adimpleti contractus)* are burnt deeply into the hypothalamus of civil law practitioners. This may serve as one reason for Toni Schnyder's attraction to issues of contract law. For as long as I had the pleasure of knowing Toni (which is the better part of 20 years), he was attracted to the sources of the legal building blocks. For some, this may seem «démodé», while in fact it shows the genuine affection for the law. Trends come and go, but the basic legal institutions will stay.

Or maybe not? This short piece highlights some features and issues of the newly emerged epiphany of contract law: the smart contract. The title is the program and the name is the claim: Dumb contracts are «passé», smart contracts the (bright) future. But how do they work, and how smart are they really? And will the machines make lawyers redundant, finally, as they made redundant the factory worker and the banking analyst? After an attempt to define the characteristics of smart contracts and to introduce the required infrastructure, I will try to sketch out a possible application in the insurance sector.

II. Notion

Paper contracts are a dying species. For a long time, contracts have been concluded and stored digitally. With due delay, the law has accepted the notion that digital contracts are valid even where certain formal requirements must be met. Through authentication and integrity infrastructure, digital contracts have become trusted and non-repudiable. Today, digital contracts are part of the legal mainstream.

Things do not stop here. The notion of smart contracts has been around for decades. As we shall see, smart contracts are many different things: They are (i) an automation process; (ii) a software script or program; and (iii) the means by which blockchains or alternative ledger technologies will finally come into the mainstream. Smart contracts are designed as a mechanism that acts as an efficient and trusted middleware in financial transactions. Smart contracts might even, though this is still contentious, be contracts after all.

Granted, there is no clear-cut definition of «the» smart contract. This is due to the fact that the very notion itself has evolved, and so has the technological framework in which smart contracts are intended to operate. The notion is fluid with some attributes crystallized over time.

The term «smart contract» has been coined by Nick Szabo back in 1994. In a seminal article, he defined a smart contract as a «computerized transaction protocol that executes the terms of a contract. The general objectives are to satisfy common contractual conditions (such as payment terms, liens, confidentiality, and even enforcement), minimize exceptions both malicious and accidental, and minimize the need for trusted intermediaries. Related economic goals include lowering fraud loss, arbitrations and enforcement costs, and other transaction costs.»[1] Nick Szabo saw smart contracts as improving execution of the four basic contract objectives, which he described as observability, verifiability, privity and enforceability.

Or to put it more bluntly: Smart contracts are generally considered to be a computer program that regulates and automates the relationship between two or more parties. To be-

1 Nick Szabo, Smart Contracts, <http://szabo.best.vwh.net/smart.contracts.html> (downloaded on 4 December 2016).

come valid and effective, a basic consent of the parties relating to rights and duties needs to be translated into machine-readable (binary) language.

But why would we need a smart contract after all? Nick Szabo explained: «The basic idea behind smart contracts is that many kinds of contractual clauses (such as collateral, bonding, delineation of property rights, etc.) can be embedded in the hardware and software we deal with, in such a way as to make breach of contract expensive (if desired, sometimes prohibitively so) for the breacher»[2]. Seen from this perspective, smart contracts are nothing else than an instrument to reduce transaction cost resulting from, *inter alia,* conclusion, breach, default or enforcement.

III. Classification

How exactly this «translation» from words into machine code should be qualified by law is open to discussion. Given the self-executing character of smart contracts, one could distinguish formation of the contract (which is done by men) and consummation (which is done by machines). Accordingly, a smart contract would not be a contract at all, but only an automated closing equipment.

Seen from another angle, the translation (programming) could be interpreted as the recording of the parties' consent in the agreed form (which is computer code; Art. 16 Swiss Code of Obligations (CO)). This form requirement affects the validity of the contract, or in other words: what is not in the code, is not part of the agreement.

Yet another approach would understand the binary recording of the original (human) consent as a novation of the entire contractual arrangement, although a novation is not assumed by law (Art. 116 CO). Seen from this angle, a smart contract would not just be an ancillary deed which records an agreement, but a contract in its own right. This would, *inter alia,* exclude the notion of «translation errors» (Art. 24 CO) which would potentially void the smart contract. However, if the novation was not successful, we would end up with an old-fashioned oral agreement – which may not exactly be what the parties anticipated at the outset.

In the end, it will still be up to men (at least for the time being) to shape the smart contract and to decide on issues such as form, execution, novation or exclusion of challenges to the validity (Art. 19 CO). Such decision should be guided by some overarching principles which are pivotal for the success of smart contracts. Such principles would include: (i) «self containment» of smart contracts; (ii) exclusion of challenges based on extrinsic factors; (iii) standardization of contract terms; (iv) use of defined, non-ambiguous terms

2 Nick Szabo, Formalizing and Securing Relationships on Public Networks, First Monday Internet Journal, Volume 2, Number 9, 1 September 1997, available at: <http://firstmonday.org/ojs/index.php/fm/article/view/548/469>.

(as in any other programming language) which defy interpretation; and (v) agreement of clear rules coupled with enforcement mechanics once the obligations fall due.

IV. Features

Smart contracts are by definition self-executing, once certain pre-defined conditions are met, and designed to allow for the exchange of (digital) consideration and/or objects in the real (read non-digital) world. In other words, once the conditions agreed upon by the parties and implemented in the code are met, this code will perform certain actions, automatically, with no human interaction needed or tolerated. For example, in a bet between two players on the result of a soccer match, the smart contract will automatically transfer the loser's stake in electronic currency to the winner, after having autonomously checked the results on a (previously agreed) news site; beside the trusted information regarding the outcome of the game (sometimes dubbed as the «Oracle»), no (other) intermediaries are needed and the parties cannot intervene in the process.

In contemporary legal theory, implementation is not an attribute of the contract itself, but its consequence, a subsequent step. Indeed, frequently, contracts are signed before the parties spend too much thought on the closing mechanics. This is one – and not the rarest – reason for later disputes. By forcing the parties to anticipate the consummation of the contract, smart contracts avoid such disputes. They combine formation and consummation in a single step, as illustrated by the good old vending machines. After selection of the merchandise and insertion of the coins, no other action is necessary, automation replacing the need for any other form of communication of the parties' intent.

To be able to operate, smart contracts regularly rely on a digital infrastructure and digital methods or payment. Thereby, smart contracts are able to ensure the reliability and security of any transaction and, therefore, gain the parties' trust. By extension, many hope that smart contracts will eventually reduce the need for middlemen, banks, lawyers and eventually also judges, because they will simplify day-to-day transactions and provide direct access to payment and collateral at no extra cost. Expensive instruments such as pledges or escrow accounts would thus become dispensable. For the legal practitioner, this may sound less beneficial. But again, as the legal instruments change, so will the training and the skills of the lawyers. The «smart» lawyer may well be a multi-disciplinary practitioner combining legal skills with IT and programming knowledge.

While not all contracts are prime candidates for conversion into smart, self-executing code, there are several types which literally lust for unlocking their smart potential. In the first place, where money is invested or lent according to standardized terms, smart contracts are the instrument of choice. The concept would also work for simple (and standardized) sale and purchase, auction or lease agreements. The benefits (in terms of transaction costs) are always obvious where a great number of agreements are concluded on the same or similar terms – as is the case in the insurance industry.

For example, a bad weather allowance policy would simply count the days of rain or cold weather from a trusted site in any defined period and pay out compensation as soon as the agreed threshold has been met or exceeded. Or a crop-hail insurance would base payments on radar or satellite weather data for a certain region. There may even be a case for third party liability insurance in the automotive industry. «Smart» cars (equipped with drive recorders and other devices) will offer detailed information on driving conditions and the degree of damages in case of a crash, allowing at some point to define and settle insurance claims automatically, without the interference of a claims adjuster (which may be needed for some time as Oracle).

V. Blockchain

Historically, smart contracts have been closely linked to the *blockchain* technology. While such linkage is not compulsory, it does not come as a surprise either. Both ideas root in libertarian concepts which ultimately aim at wrestling the control of the markets currently held by governments and central banks and vest it in the «people» (whoever that will be). The blockchain is private, decentralized and denationalized. Thereby, it would offer a (but not necessarily «the») suitable infrastructure for the implementation and enforcement of smart contracts, isolated from political ideologies and the zeal of central bankers.

In a nutshell, the blockchain is a database. More specifically, the blockchain consists of distributed, decentralized transaction ledgers which are operated and maintained in a peer-to-peer environment and stored in each node (operated by so called miners). Transactions are collated in blocks which are then hashed (i.e. given a cryptographic fingerprint) in pairs and incorporated in the chain of prior blocks. Any change of a transaction will also change all subsequent blocks. Therefore, the blockchain is (at least in theory) immutable, i.e. tamper-proof.

The blockchain came with the rise of Bitcoin, currently still the most famous and colourful cryptocurrency. Traditionally, money flows through a widely centralised system involving banks and clearing houses. Shortly stated, these institutions ensure, among other things, that one person cannot spend the same money twice. For Bitcoin to thrive, similar guarantees were needed. However, the existing financial structures were not capable of fully implementing cryptocurrencies. Then along came the blockchain. All transactions are recorded on the blockchain, forever, and, because it is distributed, i.e. shared with all its users, everyone is informed in real time of what is happening on the blockchain. Simply put, the blockchain started off as a platform for cryptocurrencies to flow, yet with little or no risk of theft or fraud and without having to go through banks and clearing houses. Now, blockchains can be used to administer much more information than solely cryptocurrencies. Databases which convey public trust or facilitate private transactions – such as land registers, company registers or vehicle registers – could be transposed onto a blockchain.

Less abstractly, the blockchain basically works as follows: To start with, each block contains information, such as information on a transaction between parties. Everyone on the blockchain can see that there is information in the block as well as the general description of this information, but cannot access this block which is encrypted.

Then, there is the chain: because the ledger is distributed, no block can single-handedly be deleted. The blocks are therefore perpetual. This means that when a new block is created, approved and «hashed», older blocks do not disappear. Each block is thus chained to a previous one and all this information is synchronized on all the copies belonging to each participant to the blockchain and encrypted through the rules of a so-called «consensus algorithm». This seemingly simple process is essential. It means that it is not possible to lie to the blockchain. If one attempted to do so, the new «block» would simply not make its way into the chain, because it would not be approved as it would be in contradiction with the other, valid, blocks. From this angle, the blockchain incorporates elements of a title chain on the back of share certificates or in the land register as concerns ownership of a real property.

Because it is shared, the blockchain could help its users avoid having to enter the same information several times in different databases. For instance, each time an insured person changes insurance company, the blockchain will still contain all the relevant information as these companies will use a blockchain instead of their individual databases. Also, since it relies on a shared structure, it is extremely difficult to tamper with, because every user has a copy and any changes have to be approved. This gives security to its users. Therefore, the blockchain reduces infrastructure costs and tampering risks associated with a centralised database, eliminates or greatly reduces transaction fees and the reliance on middlemen and does so in a secure fashion.

Where do smart contracts come into the picture? Smart contracts, conceptually, predate the blockchain by many years. This is paradoxical, as in a way they are to blockchains what fish are to the sea. With the development of the blockchain, smart contracts found the infrastructure needed to swim. There are and will be other, more sophisticated infrastructures available in the future, but for the time being, smart contracts predominantly rely on the – or rather, on a certain variance of a – blockchain.

Moreover, smart contracts may be the application that brings the blockchain idea out of the libertarian corner and into the business mainstream. Thanks to cryptocurrency (Bitcoin, Ether, etc.) and to assets being entered into the blockchain (IP rights, ownership titles, etc.) and owing to the fact that they can potentially control real world objects (see below on the Internet of Things), smart contracts will be able to automate transactions. This in turn offers the prospect to contracting parties that the terms of their contract will always play out as they have agreed and that they will not be cheated either by their contracting counterparty or by a third party. Smart contracts could, for instance, transfer the title of a vehicle registered on the blockchain against a certain predefined amount of cryptocurrency. Because it is automatic, the smart contract mechanism also curtails the risk

of non-performance. All this is based on the assumption that machine code does not lie or cheat.

However, blockchain is no religion, just one of many technologies. There are and will be other ways to guarantee the key ingredients of smart contracts (which are, as seen above, observability, verifiability, privity and enforceability). Moreover, the blockchain is not carved into stone, but will evolve over time. As the experience with The DAO (a decentralized venture capital fund based on blockchain technology which I will not revisit here apart from a brief overview in Section IX)[3] has painfully exhibited, there is an obvious need for concepts to rectify faults.

VI. Ethereum

While cryptocurrencies (such as Bitcoin) are a pre-defined asset, there is no need for individual code to trade such an asset. Smart contracts, however, require that specific information on the parties, their obligations, time, reliance on Oracles etc. be recorded in the code. To this effect, the Ethereum platform was set up. Ethereum is a «decentralized platform that runs smart contracts: applications that run exactly as programmed without any possibility of downtime, censorship, fraud or third party interference.»[4] Not surprisingly, Ethereum – which is developed and run by a Swiss non-profit foundation – relies on the blockchain technology.

Ethereum is a programmable blockchain allowing users to draft their own contracts (with names, delivery and payment obligations, time limits, escrow obligations and many more). In a nutshell, Ethereum offers a platform with a (more or less) user-friendly interface allowing anyone to put together a smart contract. There are no inherent limits to the complexity of smart contracts in the Ethereum. Smart contracts may be programmed in a number of known existing languages and will be compiled uploaded and run in the so called Ethereum Virtual Machine, an isolated runtime environment.

Smart contracts on Ethereum use the proprietary Ether currency as consideration. Ether is a cryptocurrency (such as Bitcoin) and has a market capitalization in excess of USD 1 bn.

VII. Internet of Things

Another driver of smart contracts is the Internet of Things («IoT»). IoT refers to the idea that more and more physical objects connect to the Internet to send and receive infor-

3 See instead Emin Gün Sirer, Thoughts on The DAO Hack, <http://hackingdistributed.com/2016/06/17/thoughts-on-the-dao-hack/>.

4 <www.ethereum.org>.

mation. For instance, we can use our smartphone to connect to the central heating of our home over the Internet – or to heat a room to a specific temperature at a specific time. Many objects contain sensors allowing them to react to the environment: a smart home can turn on and off the lights as the inhabitants move from room to room. Additionally, machines can also communicate with other machines: the car can open the garage door when it nears home; the garage door can turn on the microwave for the pizza; and the microwave can power up the stove to heat the dishes.

In the automotive industry, IoT applications are increasingly used for the fleet management of large enterprises. Thereby, the fleet manager is offered a real-time picture of where the vehicles are located at any given time, mileage, fuel consumption, next maintenance terms, malfunctions, call-back options and other useful information. Beyond information, the IoT offers further possibilities: a smart lease contract could reach out to a (equally smart) car and tell it to lock its doors in case a monthly lease payment is left unsettled. And a smart insurance contract will rely on (static and kinetic) data from the collision to ascertain causation and (where relevant) fault and to release payment to the insured or (depending on the system) directly to the injured party.

VIII. Smart Insurance

The elements of a smart contract for automotive insurance may be summarized as follows. The insured party (vehicle owner) pays an amount (measured according to transparent and readily available criteria such as purchase prices, horsepower, torque or capacity) to a centralized insurer or directly into a blockchain pool. The car of the insured is equipped with a drive recorder that records such standard data as vehicle location, acceleration, velocity, braking etc., but eventually also data on the fitness of the driver, driving history, fatigue, drug abuse and others.

If an accident occurs, the insured party (assuming there is no direct claim of the injured party as provided under certain national laws) makes a claim against the insurer or the blockchain pool to be awarded an amount to cover the claim of the injured party. The entitlement as such and the amount of the claim will need to be verified by an Oracle. If cars are equipped with adequate sensors, the extent of the damages may be appraised by relying on digital evidence. The repair costs are available through a database for all vehicles – as is the case today with the SilverDAT database which is used by many dealers and repair shops in Switzerland. To the extent no reliable binary Oracles are available, the code will delegate this task to a claims adjuster who may be appointed by the investors in the blockchain pool based on transparent criteria. The claim amount is then transferred from the blockchain to the insured party without further interaction by an intermediary needed.

While this undoubtedly sounds like sci-fi, the future is not too far away. The tremendous cost savings will pave the way, despite the legal obstacles. These will include: (i) identification and authorization; (ii) privacy and data protection; (iii) fraud; (iv) errors and omis-

sions in the (not always smart) code; (v) false Oracles; (vi) consumer protection issues; and (vii) competition law.[5] At the same time, this is the bright side of smart contracts: they offer an array of legal problems which need to be solved by smart lawyers.

IX. Legal Challenges

Although smart by definition, these contracts are not immune against errors, omissions and faults. The following example highlights a case when smart contracts turn sour.

Imagine first that an insurer publicly declare that all claims will be settled within 30 days without exception and deductions. Now, as the car was stolen and the reckless thief caused damages in violation of all and any traffic laws, the agreed Oracles (such as the drive recorder and GPS) may tell the smart contract not to honour the claim by the injured parties based on gross contributory negligence of the driver. Suddenly, the vehicle owner is confronted with private tort claims. *Quid iuris?*

The easy answer would be: the smart contract is what it is. Or put otherwise: Who does not read the small print shall not complain afterwards. But this may not be the end of it. What if the smart contract cannot reasonably be understood by the customer? Shall it then be interpreted «*in dubio contra stipulatorem*»? In fact, the criteria for general terms and the attached consumer protection regulations will usually be met for smart contracts which are, by nature, standardized and scalable. But contrary to the contracts known to Eugen Huber and his peers, there is nothing to interpret with smart contracts. Again: They are, what they are (in Nick Szabo's words: «dry code»), and they are what and how they perform.

Anyway, the customer may claim that he or she was misled by the full-mouthed promises of the carrier (Art. 28 CO). Thereby, the validity of the contract would be challenged. Again, the concept of actual or «normative» consent has its obvious limits with smart contracts. Here, the issue is the degree of care the customer must observe when entering into the smart contract. If reliance on an abstract promise provided by the other party or other industry source is customary, then there is also the option of claim against the issuer of the false promise based on the breach of good faith. Needless to say that this is an uphill battle. However, given the nature of the smart contract (which is fully transparent and cannot be changed after the fact without atomizing the very idea of the distributed ledger technology), the option of suing a sponsor issuing false promises should be available to a customer who relied on such promises in good faith.

5 I will not be able to address all the above points in this short text. For an overview and some preliminary answers see Rolf Weber, Contractual Duties and Allocation of Liability in Automated Digital Contracts, in: Reiner Schulze/Dirk Staudenmayer (eds.), Digital Revolution: Challenges for Contract Law in Practice, Baden-Baden 2016, pp. 163 et seq., 180 et seq.

Secondly and quite the opposite from the above referenced situation, the contract code may in fact be flawed. Under (quite exceptional) circumstance, where all involved were unanimous and certain of what the code was to achieve, there may be a hidden bug or leakage which was not introduced by the programmers consciously, but through an error or false judgment. The famous case (which will not be reconstructed here in detail) is, of course, The DAO. The DAO incident with Ether valued approximately at USD 50 million being siphoned off, ultimately led the Ethereum community to rescind (technically: «hard-fork») the blockchain and thereby violate the holy grail of the distributed ledger technology. While «hard-forking» is not a contract remedy readily available in the Swiss Code of Obligations, it may work – similarly as contract rescission – as a pressure valve where things went definitely sour.

Thirdly, a smart contract system feeds from interfaces to the real world delivering, *inter alia,* time, location of goods, status of delivery, interest rates or account information. These interfaces may be tampered with, or the information fed to the system may just plainly be wrong. Who to blame? In legacy contracts, we would define spheres of risk and responsibility, starting with the notion of an «auxiliary» (Art. 101 CO) or the definition of transfer of risk and benefit (e.g. along the Incoterm model clauses). There are clear and obvious limits with these approaches to risk allocation within smart contract systems. The very idea that Oracles are auxiliaries of the carrier would undermine the reliance on such sources. Rather, Oracles should be qualified as either agreed facts (even though happening in the future) or as arbitral opinions *(«Schiedsgutachten»)* which may not be challenged except if it can be proved that the source was either biased or tampered with.

Beside the issues of contract law, the heightened public apprehension (although not always compatible with private behavior) for privacy issues will put limits on smart contracts. Firstly, while information on the parties may be anonymized or pseudonymized, the distributed ledger is by definition public. If parts of the smart contract are kept private between the parties, i.e. not accessible by the miners or the public at large, the very function of the blockchain (which is to provide an observable and verifiable proof for the existence of facts and legal obligations) is curtailed. The solution may be to encrypt (e.g. through a public-key infrastructure) or obfuscate personal data as well as certain commercial secrets.[6] However, this will in turn cause both technical and «systemic» issues as the parties will have to release their private keys in case of a dispute.

Secondly, by their inherently public nature, smart insurance contracts may also lead to an exchange and ultimately to the convergence of commercial and legal terms of the carriers. This will harm competition and draw the attention of competition authorities. Again, these issues may be overcome with encryption or obfuscation, but only at a certain cost.

6 For ways and issues of encryption and obfuscation of smart contracts see Vitalik Buterin, <https://blog.ethereum.org/2016/01/15/privacy-on-the-blockchain/>.

Thirdly, criminal law (protection of business secrets, blocking statutes, banking secrecy) and regulatory constraints will prevent disclosure of certain information to the public at large or to the community of the miners.

In light of these issues, the emergence of private or proprietary blockchains appears unavoidable. In private blockchains, access permissions are controlled by a central authority.[7] Also, proprietary coins may be issued by the private blockchain «owner». This is a step away from the idea of a decentralized transaction ledger and may thus be considered contrary to the libertarian tradition which refuses control by central authorities, albeit government agencies or private enterprises. In a way, this discussion reminds me of the cloud computing discussion entertained some ten years ago. There, the very notion of a «private cloud» was dismissed by the cloud (i.e. grid) community and the large cloud service providers. Today, private clouds are omnipresent. Sometime, heresy is unavoidable to leverage a great idea.

X. Outlook

If smart contracts in general (and smart insurance contracts in particular) were to work in practice, some of the legal approaches to contract interpretation and contract challenge must be revisited. Smart contracts are «dry code» (Szabo), and they cannot be handled the same way as «wet code» (which is man-made and interpreted by the brain rather than by a machine). If dry code is challenged as general contractual terms are being challenged nowadays (i.e. for unexpected terms or for non-equivalent allocation of rights and duties), smart contracts will live a short life. As The DAO case has illustrated, challenging and «rescinding» dry code transaction will mean to re-write («hard-fork») the blockchain which, if done at regular intervals, will – as with laws and regulations – end trust and confidence in this technology.

Financial institutions have been the first to explore the potential of the blockchain technology in general and smart contracts based on such technology in particular with a view to reduce costs, increase speed, improve their competitiveness etc.[8] Other industries will follow suit. However, this will require the technology to evolve, to leave the libertarian realm and come into the business mainstream as was the case with other disruptive technologies such as the Internet, artificial intelligence or cloud computing. With the emergence of smart contract, albeit slow, lawyers will not lose their jobs. Smart contracts will

7 Different types of «consortium» and «private» blockchains are discussed by Vitalik Buterin, <https://blog.ethereum.org/2015/08/07/on-public-and-private-blockchains/>.

8 E.g. JP Morgan, Credit Suisse Among 8 in Latest Bank Blockchain Test, available at: <http://www.coindesk.com/jp-morgan-credit-suisse-among-8-in-latest-bank-blockchain-test/>; UBS: Effizienzgewinn dank Smart Contracts, about UBS's foray into the smart contracts/blockchain structure, available at: <http://www.finews.ch/news/banken/24603-alex-batlin-blockchain-smart-contracts-jerry-cuomo>.

(as much or little as this is the case for the Internet) not operate in a legal vacuum. It will be the noble task of the legal profession to find new solutions with old instruments and tools – as has happened many times before.[9]

9 See Walter Blocher, The next big thing: Blockchain – Bitcoin – Smart Contracts: Wie das disruptive Potential der Distributed Ledger Technology (nicht nur) das Recht fordern wird, AnwBl 8 + 9/2016, 612 ff., 618.