Predrag Cvetković,*[*] *LL.D.,
Full Professor,
Faculty of Law, University of Niš

ОРИГИНАЛНИ НАУЧНИ РАД
10.5937/zrpfn0-28637

UDK: 347.4:[004.7:336.74
004.7
336.74
Рад примљен: 30.09.2020.
Рад прихваћен: 02.12.2020.

LIABILITY IN THE CONTEXT OF BLOCKCHAIN-SMART CONTRACT NEXUS: INTRODUCTORY CONSIDERATIONS[**]

Abstract*: Blockchain technology becomes relevant in economic exchange as it lowers costs and contributes to cost-efficiency and effectiveness of economic transactions. The key quality of Blockchain lies in ensuring the authenticity of digital data: trust in the traditional legal relationship has been replaced by digital verification of data in blocks. As an important phenomenon, Blockchain calls for legal answers on the issues arising from its application. An example of this development is the legal regime of smart contracts. A smart contract is a transaction in which any rights and obligations of the contracting parties are programmed in a code. Being the result of Blockchain technology application, such a contract implies the need for trust between the contracting parties. As a legal phenomenon, Blockchain (smart contract) technology raises the issue of liability for performing contractual obligations. Smart contracts can minimize certain contract risks and additionally simplify contract execution. They are immediately put into effect, without the need for any further interaction between the parties. The essential components of smart contracts are the digitally verifiable data and the automatic performance of legally relevant actions based on digitally received and processed information. All of the enlisted issues are important for proper understanding of liability of Blockchain actors.*

Keywords: *Blockchain, smart contracts, liability, Distributed Ledger Technology, Ricardian contract, "If This, Than That" principle.*

[*] pepi@prafak.ni.ac.rs

[**] The paper was presented at the International Scientific Conference "*Responsibility in the Legal and Social Context*", held at the Faculty of Law, University of Nis, on 18 September 2020.

1. Introduction

More than a decade ago, in one of the publications on cryptography, a group of unidentified authors (known under the pseudonym Satoshi Nakamoto) introduced the concept of a Blockchain-based contract (Ducas, Wilner, 2017: 544). The document proposed introducing a version of electronic money (bitcoin), which uses cryptography to allow direct peer-to-peer (P2P) payments to eliminate the participation of intermediaries in economic transactions.[1]

The development of information technologies influences all areas of human existence. One of the key breakthroughs in this regard is the emergence of Blockchain technology (Cvetković, 2020: 127-144). Blockchain technology is becoming relevant in energy production, health system, education, financing, public service management, logistics, and transport. The impact of this development is reflected in the legislative efforts, aimed at:

a) regulating the Blockchain-related processes;

b) standardizing the terminology used;

c) indicating the method for resolving disputes arising from the application of Blockchain technology.

2. General Features of Blockchain technology

2.1. Distributed Ledger Technology (DLT)

The development of information technology entered a mature phase when it was possible to transfer files from one to two or more computers, which boosted the power of computer networks. The so-called Metcalfe's law stipulates the premise that the effect of a computer network is proportional to the square of the number of connected computers (nodes). A computer node is an active electronic device which is connected to a network and enables the sending of information through communication channels to a computer network.

The term "Distributed Ledger Technology" (DLT) was first used in a Report prepared by an expert group for the United Kingdom Government. DLT is defined as a type of database that extends to multiple different locations, countries, or institutions, and is typically public. The data are stored one after the other in continuous records; new data are added when the participants reach a consensus (UK Government Office for Science, 2016: 17).[2]

[1] Peer- to- Peer payment is the electronic transfer of money from one person to another through the use of a payment application without intermediaries.

[2] UK Government Office for Science (2016): Distributed Ledger Technology: beyond block chain. A report by the UK Government Chief Scientific Adviser, London, UK, 2016.

DLT is associated with the modern meaning of the term "document". The starting point is the following: the method and security of data verification are more important than the formal characteristics of the document that contains the information. Access to information and prevention of altering it (by protecting the "integrity" of information) are more important than the document itself. The essence of the document is that:

- the content of the information it contains is constant and stable,
- it is possible to copy or transfer information to another medium (in the context of Blockchain technology, it is a computer on a network) so that it remains unchanged.

In case of DLT, data verification/validation occurs automatically through an information system based on cryptography and data protection. The information is approved after verification by the participants in the network (nodes, i.e. participants behind the computers that constitute the network) who are authorized to perform data verification.

2.2. Blockchain Mode of Operation

"Blockchain" is a compound of the words "block" and "chain". It is a concept based on the use of a cryptographically protected chain of transaction blocks. Transactions are packed into blocks, and blocks are tied into a chain. Blocks are bound cryptographically, through a hash function[3]: the contents of a block cannot be changed without changing the contents of all other blocks preceding it. Namely, each block is bound to the next block using a cryptographic signature. This allows the Blockchains to be used as a digital ledger which can be shared and verified by anyone with the appropriate permission to do so.[4]

A block consists of a title and transaction data.

A title contains:

[3] The term *hash* ("hash value") comes from mathematics. It refers to a short string of a fixed length which represents the abbreviated form of a long string (checksum). A cryptographic hash value is used as a security mechanism in Blockchain technology. If a character has been changed in the original value (i.e. in the original content of the data recorded in the Blockchain), the corresponding hash value also changes. The hash value is used to compare two character strings (checksums) with one another in order to determine whether they are identical. On the Blockchain, the data record is converted into a hash value and stored inside of the block. New data shall be in the form of a block with hash value, taking into account the hash values of other data in the previous blocks (which are already part of the "chain"). If the hash value is not in accordance with the data contained in the previously inserted blocks, it cannot be verified; consequently, it cannot be part of the Blockchain at stake.

[4] See more infra in this Part.

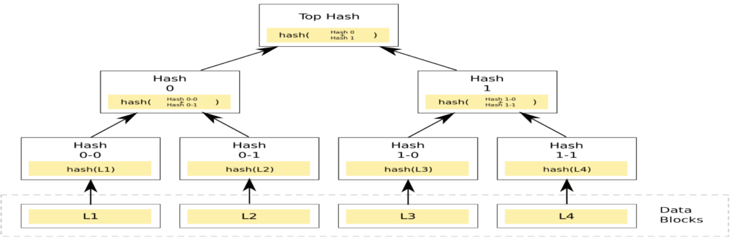
• references to the previous block in the chain, i.e a short combination of letters related to a certain set of data (hash).

• a time stamp indicating the time the block was entered into the "chain" of blocks, and

• a hash tree or "Merkle tree" which lays out all transactions included in the block.[5]

In Table 1, nodes L1 to L4 are external nodes (users), which are the point of further branching. In the Blockchain context, external nodes are the points for adding other blocks.

Table 1: Merkle Tree Concept



Source: Hash Tree, illustrated by David Göthberg, English Wikipedia, 20 August 2005

Including hashes in the block title enables the search for transactions through the hashes as their recognition signs; thus, there is no need to read all the data included in the Blockchain. In the search, the title and branches of the Merkle tree are automatically readable. This practice is analogous to searches in a traditional hard copy ledger; the title and data from the ledger are obtained by reviewing the contents of the ledger and page references. The only difference is in the search method; data from the Blockchain are searched automatically; in traditional ledgers, it is a physical search. However, unlike traditional hard copy ledgers, DLT functions as a decentralized system; each participant has its own

[5] The concept of *hash tree* is named after Ralph Merkle, who patented it in 1979. In cryptography, "Merkle tree" denotes a network structure in which each external user (called a "node") is marked with a hash; any other node that branches further contains the particular hash marking all sub-branches arising from that other node. Hash-branches enable efficient and secure verification of the contents of voluminous files.

copy, or part of the register, identical to the copies of other participants (nodes). It means that everyone has access to all the data included into the digital ledger.[6]

The process of data verification and ensuring the consent of other participants in the Blockchain concerning the entry of new blocks of information is performed automatically. After the consent (approval) is obtained, the new blocks are registered in the chain and cryptographically secured by those participants who have carried out previous transactions by tying new blocks to previous blocks. The chain generated in this way is difficult to change. It is virtually impossible to destroy it due to the large number of copies of the same data (available in different blocks); destroying one copy would require a simultaneous and effective attack on other Blockchain participants; (as for this feature, the Blockchain design is similar to ARPANET; the latter is the forerunner of the modern Internet created in order to eliminate the loss of data in case of infrastructure network damage) (Leiner, *et al*, 1997:3)

2.3. Types of Blockchain

The most important typology distinguishes between public and private Blockhains.

The disruptive impact of the Blockchain concept is attached to the public Blockchain. The public Blockchain is fully accessible to everyone; it is based on the so-called open source code[7] and the software solutions are fully accessible. Anyone, without any personal or territorial restrictions, may install the appropriate software required for the operation of the public Blockchain on the device, record in whole or in part a fragment of the files, and make their copy available to other users. Anyone can request the addition of any block of information (transaction) to a chain of blocks. However, the transaction will be accepted when other Blockchain participants have agreed to it. No access

[6] Verification of digital data by tracing them through blocks is identical to a hard copy ledger; blocks function analogously as bookkeeping inputs of a digital bookkeeper. Blocks are functionally equal to sheets of paper, used by all participants to enter their transaction and to sign it. In doing so, they grant authorization to all previous paper transactions. This process continues as long as there is space on paper available. When the sheet is filled in, it is secured with a stamp; new transactions are recorded on a new sheet of paper; once filled in, it is linked to the previous paper (secured with the signature and stamp at the boundary between the first and second paper). Functionally speaking, the identical activity is conducted in the framework of Blockchain technology.

[7] Open source code is freely available to users; anyone can download the source code, modify it and distribute its modified version in an unlimited number of copies. There are no license fees or any other restrictions. A more detailed and technologically developed definition is given at the Open Source Initiative website: https://opensource.org/osd (accessed on 01. 05. 2020).

rights are required and no single entity manages the Blockchain. With no one in control of the network, public Blockchains are genuinely decentralized systems. New blocks are verified by the entire network. It is not necessary for a separate trusted party to monitor the operations. Accordingly, public Blockchains are trustworthy. Anyone who wants to change data on the Blockchain needs permission from other participants. As a result, manipulating data on public Blockchains is next to impossible.[8]

From a technical perspective, a private Blockchain is based on the same technology of linking blocks into chains as it is the case with a public Blockchain. However, there is a crucial difference: private Blockchains are owned by a central entity (one or more units). The owner can decide who can join the network; his function is analogous with the one of a central network administrator. A private Blockchain is used when the network contains confidential information; consequently, activities in the ledger require authorization by the administrator. The capacity of a particular person to use a private Blockchain usually arises from an agreement concluded between the users themselves. Private Blockchain is usually (but not only) used in projects and agreements of a lucrative character. A private Blockchain is not genuinely decentralized; actually, it is just a cryptographically secured distributed ledger. To carry out transactions, the participants in the network are still dependent on a third party - the Blockchain administrator.

2.4. Blockchain as a Trustless Concept

Blockchain creates a trustless system which may function without the need for mutual trust among contracting parties, The basic idea of the public Blockchain is to overcome the traditional aspects of trust that play a central role in everyday business life. The parties enter into contracts with partners who are expected to comply with the agreements. This expectation stems from the contracting party's reputation, data from public registers, or personal conviction. Trust plays a central role in traditional contract law. However, trust is not a prerequisite for entering into a Blockchain transaction. Due to the technical possibility of storing unaltered data, in a decentralized and distributed manner, there is no need to have trust in another party; trust is functionally replaced with reliance on the technology that Blockchain is based on. Ultimately, the Blockchain concept brings about a paradigm shift in contractual relations; trust in the human is replaced by reliance on technology. In case of a public Blockchain, trust is ascertained

[8] Typical examples of public blockchains are Bitcoin and Ethereum. For more on Ethereum, see infra in footnote 13.

by numerous operators of the Blockchain network nodes; consequently, the Blockchain as a system is not dependent on a single participant.

The result of this paradigm shift is that costly intermediaries (such as banks) are no longer required. However, this result is not fully reflected in case of private Blockchain. The participation in a private Blockchain is subject to permission or fulfilment of certain conditions. It is operated by a limited network of participants, according to mutually defined rules. Trust is based on the closed community of participants who initiated the private Blockchain. Therefore, one can no longer speak of a "trustless" system, given that a single player (the administrator) is the one who is trusted. Consequently, in case of a private Blockchain, reliance on technology is of secondary importance.

From the present-day perspective, it is difficult to assess whether a "trustless" system can also have an impact on the basic principles of contract law. Given the rapid development of technology, it is highly unlikely for the time being. It should be noted that the trust aspect is an essential element in interpreting contracts. It is true that the automation of contracts by means of Blockchain technology (particularly in case of smart contracts) reaffirms the *pacta sunt servanda* principle. However, it cannot fully exclude the need for interpretation of the terms which are not based on the principle "If this, than that".[9]

3. Blockchain and Smart Contracts *Nexus*

Blockchain and smart contracts are two different technologies that are closely correlated.

A smart contract is a Blockchain-based computer program (hereinafter: a code) that authenticates, enables and implements the contract norms contained in program code. It is based on a cryptographic process enabling the execution of contracts once the terms and conditions contained in the code have been met. In compliance with the contracting parties' agreement, a smart contract automatically fulfils the envisaged obligation. Once the smart contract (in the form of a program code) is entered into the Blockchain, the contract can only be executed in line with the loaded program code. The main goal of applying Blockchain in the context of smart contracts is to make the contractual relationship more efficient and economically viable, with fewer opportunities for errors, delays or disputes.

The term "smart contract" dates back to 1996, when Nick Szabo defined it as a series of digitally recorded promises and protocols, by means of which the

[9] For more about smart contracts and the principle "*If this, than that*", see infra in parts 3 and 4.

parties keep these promises without the involvement of intermediaries.[10] Szabo is the author of the canonical definition of a smart contract: it is a computer protocol (program) for carrying out a transaction in accordance with the terms of contract. The main goals of smart contracts are to: ensure the performance of contractual provisions (e.g. payment terms, surety, confidentiality, execution), and minimize the need for "honourable" impartial intermediaries. The basic idea of a smart contract is that many contract clauses (e.g. surety, advance payment, authorization specification, etc.) can be embedded into the code and uploaded into hardware, thus ensuring that the costs of contract breach are so high for the infringer that it makes the breach unlikely (Szabo,1996:1).

The key feature of smart contracts is that they can be presented in the program code and executed by computers; hence, they differ from traditional contracts usually established through negotiations, written documents and conclusive actions. Smart contracts are self-implementing and self-executing computer programs based on a program algorithm (Lauslahti, Mattila & Seppälä, 2017:2)

Self-service (vending) machines may illustrate the operation of a smart contract; these machines are computerized, thus avoiding the interaction and participation of a third party (intermediary); they are programmed to deliver the product without the need for human when certain conditions are met (i.e. when money is put into the machine slot).

The key features of smart contracts are as follows:

1) smart contracts are created (programmed) by using an open source code;[11] their standardization and execution are almost cost-free, thereby reducing the contract transaction costs;

2) smart contract potentially narrows the space for ambiguous or vague interpretations, thus increasing the efficiency of contract execution; when the parties agree on the content of the clauses, the smart contract program code executes those clauses without the possibility of breach of contract;[12]

[10] Szabo described the idea of contracts that could be read and used by humans and machines alike; however, it was not technically implementable at the time. The term "smart contract" has resurrected in recent years, with the development of Blockchain technology.

[11] See supra footnote 7.

[12] Written in a programming language, smart contracts eliminate the ambiguity of natural language. This feature limits the usefulness of the smart contract conceptual framework; namely, parties may prefer the flexibility of legally binding contracts to the rigidity of automated software. For the time being, smart contracts cannot measure up with the discretion rooted in legally binding contracts or their linguistic ambiguity. Terms such as "the best possible effort" or "force majeure" cannot be reproduced in the code. See more infra in part 4. 2.

3) smart contracts are designed to operate without intermediaries (in a decentralized format);

4) a smart contract is a self-executing program, especially in Blockchain technology, which aims to ensure that the parties perform and execute automated transactions; the execution can be based on data from the program, or it be the result of data collected from the environment in which the transaction takes place; a smart contract benefits from the security of the underlying Blockchain infrastructure (the multiple Blockchain nodes): for example, its execution cannot be stopped by individuals or groups unless this option has been specifically integrated into the code.

3.1. Smart Contract: Legal Issues

The legal effect of smart contracts cannot be disputed and their validity cannot be *a priori* denied only because they comprise “smart instructions” or because the parties’ consent is expressed in a way that is not in compliance with traditional contract law.[13] Although they have emerged quite recently, their importance has been recognized by national legal systems.[14]

[13] After Szabo had published his conception of a smart contract, the idea was embraced by Vitalik Buterin, a co-founder of the Ethereum Blockchain. In his opinion, the original Bitcoin Blockchain had a limited use in software development. In response, the Ethereum Blockchain was launched in order to enable the use of the advantages of a distributed database together with a more versatile programming language, expanding thereby the areas of application of the Blockchain technology. Ethereum is the golden standard of smart contracts. It is a software platform based on an open source code providing the ability to create and activate decentralized applications. Ethereum allows the users to run various programs on Ethereum Virtual Mashine-EVM, regardless of the programming language. This feature creates space for the development of more applications on one platform instead of building a completely new application for each specific case. Ethereum allows parties to enter into an agreement, while guaranteeing the confidentiality of the transaction. It is an illustration of the evolution of the Blockchain from a payment mechanism to an effective instrument for regulating mutual relations. The Ethereum allows flexibility as a prerequisite for programming functionality, thus overcoming the immutability of the program code. The Ethereum platform enables the creation of smart contracts that define complex obligations of the contracting parties, sanction their arbitrariness, monitor the state of contract execution, and the like. Once entered in the program code, each contract term and condition is in a stand-by mode, waiting for a ‘trigger” that it has been fulfilled. Once the “trigger” ensues, the rule contained in that condition applies automatically. Smart contracts are not a passive list of instructions enumerating the contracting parties obligations; rather, they are perceived as “autonomous agents” who execute a certain part of the program code (“smart contract”) when they receive certain information defined as a “code trigger”, which is the condition for the execution of the “smart contract” norm). See: Buterin (2014) Ethereum White Paper.

[14] Definitions of smart contracts are incorporated in the legislation of some European countries, such as two legislative acts of the Republic of Malta regulating Blockchain issues:

From the technology-neutral perspective, a smart contract is understood as a computer code of contractual significance. Thus, smart contracts can exist independently from Blockchain technology, and they are already used in a variety of ways. In relevant literature, a smart contract is defined as a combination of the following properties: (1) a digitally verifiable event; (2) program code which processes the event; and (3) a legally relevant act that is carried out on the basis of the event (Kaulartz & Heckmann, 2016: 618). In contrast to this definition, the focus of the legal definition is not on technical details but on contractual effects of computer programs.

From the legal perspective, a smart contract is a computer program that is stored in a tamper-proof manner and guarantees that predetermined action will be taken when certain conditions (defined in the code) are met. The parties define (in the form of in a smart contract, i.e. a code) their fundamental contractual obligations as well as the consequences of breaches of duty or changes to essential contractual framework conditions. At the same time, they link the code to data sources enabling the code to automatically recognize the fulfilment of the stipulated conditions. If there is a breach of duty or a change in the contracted terms and conditions, the software can automatically trigger the legal consequences attached to the contracted obligations. In contrast to traditional contracts, which often require interpretation, smart contract concept offers a high degree of legal security; its legal consequences are clear, given that the code ensures that a certain clause will be put into effect when certain conditions are met. In contrast to traditional ones, the result of a smart contract is almost "guaranteed". This is beneficial in many ways. For example, in case of service contracts, when service delivery is disrupted, the resulting claims can be processed immediately. Owing to smart contracts, transaction costs are reduced, contracts are concluded more quickly, and legal certainty in business transactions is increased. Finally, thanks to the Blockchain technology, the traceability of transactions increases transparency and verifiability of transactions.[15]

the Malta Digital Innovation Authority Act C901, and the Virtual Financial Asset Act C778. Both include an identical definition of smart contracts; these contracts are a form of innovative technology consisting of: a) a computer protocol and b) an agreement concluded in whole or in part in electronic form that is automated and executable by the executing program code, although some parts may require human input and control; it can be enforced by the traditional legal method or by using both methods. The above definition adequately reflects the essence of a smart contract and can be considered a model. See: *Malta Digital Innovation Authority Act C901* of 11 November 2018 (accessed on 1. 09. 2019).

[15] See more supra in part 2.1-2.2. Appropriate cost reductions can create new business models and markets, such as: Peer-to-Peer energy markets that use intelligent smart grids or solutions with micro payments; consumer contracts where the software can automatically carry out legally required reimbursements, etc. Furthermore, insurance providers are experimenting with products that are fully automated from contract conclusion to pay out;

The provisional legal definition of a smart contract is as follows: a smart contract is a contract connected to a computer protocol, written in a computer programming language, which automatically performs programmed functions in response to the fulfilment of certain conditions ("If this, than that" principle).[16] The concept described is not new but, when integrated with Blockchain technology, it builds the potential of smart contracts to automate and guarantee the fulfilment of a large number of different contractual obligations without the need for a central authority, legal system or external enforcement mechanism. Smart contracts potentially bring clarity, predictability and controllability, and ultimately facilitate the fulfilment of contractual obligations while reducing the risks associated with human participation (Sherborne, 2017: 3-4).

4. Liability as an issue in Blockchain-Smart Contract *Nexus:* key issues

The long-standing debate on whether human beings are responsible for the operation of machines has been part of legal discourse since the industrial revolution at least. In all jurisdictions, the answer to this question has been the same: the law will cover, and be applied to, new situations and inventions appropriately adapted to the new circumstances. However, conceptions of liability did not adequately keep pace with advances in technology. While increasingly relying on automated systems, the nearest human operators were being blamed for the accidents and shortcomings of the purported "fool proof" technology. There was a significant mismatch between attributions of responsibility and how physical control over the system was actually distributed throughout a complex system and across multiple actors in time and space.[17]

4.1. Code as Law

The metaphor "code is law" (ascribed to the Lawrence Lessig) rests on the functional equality between law and a code, given that a code controls behaviour just as law does (Lessig, 2000:1). Code design and structure define the users' freedom; thus, a code determines what users can and cannot do, and what they must and must not do when using it. Hence, some legislatures defined that smart

an example of this is flight delay insurance, which is linked to a publicly accessible air traffic database, which in turn initiates the claims settlement process as soon as a relevant delay has been detected from reliable database.

[16] See more infra in the part 4. 2.

[17] For example, while flight control increasingly shifted to automated systems, responsibility for the flight still rested with the pilot.

contracts are as legally effective as the traditional ones.[18] Such definition could lead to an extreme view: if a code defines what is the 'law', anything under the coded design could be considered as a legal rule. However, the institutionalized law-making bodies are vested with the power to make rules of the specific legal system; therefore, code is *not* law.

Furthermore, the unconditioned "code is law" approach could lead to embedding the *ex ante* normativity into the code. Namely, each code depends on the subjective value-judgments (social, economic, philosophical) of its designers. The importance and influence of this dependence is magnified when the code is widely adopted. Fixed in the technology parameters (code, access conditions), it causes systemic effects. Unlike the political decisions or legal documents (e. g. contracts), computer codes are often locked for future changes. Consequently, the approximation of values embodied in the code has to be conducted before the code concerned becomes operational (*ex ante)*. The legitimacy of values embedded in a code shall be conducted in the production phase, given that a code is often irreversible; once it is developed and applied in society, it is difficult to change or remove it. Law shall be able to secure the effective and efficient evaluation of the code. For the time being, there is not enough argumentation for a reliable and credible approximation of values embodied in the code.

4.2. *De lege lata* Limitations in the Application of Smart Contract: "*If This, Than That*" format

In addition to its numerous positive aspects, the automated execution of smart contracts has some disadvantages from a legal perspective. For example, it is impossible for the party to withdraw from the contract once it has become effective. Furthermore, it is also not possible to adjust a smart contract if the

[18] For example, the Electronic Transactions Act of the State of Arizona defines Blockchain technology and specifies some of its consequences. Article 5 of this Act stipulates that Blockchain technology is a form of application of the DLT concept, which uses a distributed, decentralized, shared and duplicated database that can be public or private, with or without permission, run by a tokenized crypto economy or without a token. The data in the database are protected by cryptography, immutable and suitable for control, and provide uncensored accuracy. Article 5 allows smart contracts to be used in business relationships. Therefore, it is impossible to ignore the effects of contracts only because they are concluded as "smart contracts". Furthermore, notwithstanding other regulations, the data provided by using Blockchain technology are considered equivalent to other data whose integrity is protected in other ways. For example, this principle applies to a contract for the transfer of property rights. See: *An Act amending Section 44-7003, Arizona revised Statutes; amending Title 44, Chapter 26, Arizona revised Statutes, by adding Article 5; relating to Electronic Transactions;* (accessed on 01. 08. 2019).

circumstances affecting it have changed. In addition, there is also no possibility of intervention in the event that the code subsequently turns out to be faulty.

The most illustrative disadvantage of smart contracts in the legal discourse is their (current) inability to provide the necessary flexibility of contractual framework. As previously noted, a smart contract is a contract connected to a computer protocol, written in a computer programming language, which automatically performs programmed functions in response to the fulfilment of certain conditions ("*If this, than that*" principle). A single block in the chain is created without errors if it can be successfully linked with other blocks in the Blockchain. In this sense, the execution of a smart contract transaction cannot be incorrect; it is either successfully processed or not.

Another debatable issue is whether the legal flexibility embodied in legal standards can be transposed into a program code. By analogy with vending machines, where the execution relies on mathematical calculation (i.e. whether a sufficient amount of money has been paid to deliver the goods), smart contracts also rely on a precise and predefined execution logic. Yet, it raises the issue how some legal concepts (e.g. "reasonable conduct" or "best efforts"), which are used in traditional contracts to provide flexibility, may be transposed into the program code. The transposition of such concepts into the code by reducing them to a code algorithm may be difficult (if possible at all).

One of the ways to resolve this issue is to create the so-called hybrid forms of contract that can be "read" by both machines and humans. A typical example of such hybrid contracts is the so-called Ricardian contract. It was first introduced by the financial cryptographer Ian Grigg in 1995 (Grigg, 1996: 1, *passim*). The Ricardian contract is readable both by people (as any traditional paper contract) and by machine (a software program). The Ricardian contract does not automate the given elements of the agreement through the application of the program code. Instead, its goal is to provide flexibility for agreements in textual form, while providing them a certain degree of code identity; namely, the Ricardian contract converts an agreement in textual form into the program code, ensuring compliance to the extent which does not affect the flexibility of the norms contained in specific agreements. The ultimate result of this process is that program code complements rather than replaces agreements in textual form (by applying the formula "more rights/duties- less software").

The text of the contract which is not fully coded should be formulated in a way that corresponds to the minimalist semantics of the code. The described minimalist semantics enables the program code to guarantee the integrity of the information contained in the code (information contained in the contract itself and converted into a programming language) and verification of its origin. The text of the agreement contains all the possibilities and nuances of the language used by the contract law to meet the parties' requirements. In this way, the immutability of the program code is combined with the flexibility of expression. Flexibility as a possibility of choice is materialized in the agreement itself; on the other hand, the program code ensures the immutability of information. It remains to be seen to what extent this approach is practically useful from a technology perspective.

4.3. Responsibility for Legal Compliance and Liability Standards

In decentralised networks, it can be burdensome to identify the actors liable for legal compliance (i.e. to define the so-called "regulatory access point"). Identifying a regulatory access point is, however, more complicated where there is no centralised legal entity responsible for the network. It is among the most important regulatory issues to have emerged in relation to Blockchain and smart contracts (European Commission, 2018:47).[19]

There are two types of liability for malfunction of the code and non-fulfilment of a smart contract: 1) strict liability for any fault in a code, and 2) liability based on the reasonable-care standard.

Under the strict liability standard, code designers may be held liable for any defect in the code which has been used to make the system operative. As a result, the costs of code developers would be so high that innovation would not be financially viable. Hence, subject to the prevailing application of strict liability principle, it is likely that any Blockchain/smart contract development would be disincentivized (European Commission, 2018:47).

An approach that limits liability of developers by establishing certain standards of conduct could help safeguard and promote innovation and risk-taking. Hence, in order not to make the costs of innovation too severe, the legislature might eventually develop the liability standard focussing on reasonable care and best efforts. For instance, it may be expected that industry will do its best to ensure that code-based systems are secure against cyber intrusions; yet, perfection as such may not be expected and a lack of it will not be legally sanctioned (European Commission, 2018:48).

5. Conclusion

Blockchain technology is increasingly relevant in different fields (energy, healthcare, education, financing, public services, logistics, transport). The ad-

[19] European Commission (2018): Study on Blockchains: Legal, governance and interoperability aspects (SMART 2018/0038), European Commission; (accessed on 1. 05. 2020).

vantage of Blockchain technology lies in ensuring the authenticity of digital data: trust in the classic legal relationship has been replaced by a mechanism for verifying data in blocks without the participation of a third party. The potential of Blockchain development is clearly reflected in smart contracts. Smart contracts are Blockchain-based computer programs that authenticate, monitor, and implement contractual obligations which have been converted into a program code. The code automatically performs the obligation in accordance with the terms and conditions that the parties have decided in the agreement. Smart contracts are becoming a reality; therefore, an adequate legal response is required. But, the requisite response is highly specific because it lies on the brink between law and technology, two fields whose interaction has been exclusively technical for most of the history of their coexistence. Their intersection and overlapping open the plethora of new issues and demand the rephrasing of the old ones. As an example of this interaction which is as much important as it is intensive, Blockchain and smart contracts *nexus* creates a new reality and offers experience for further elaboration on the issue.

In terms of liability in the Blockchain/smart contract discourse, the basic question is whether legal flexibility embodied in legal standards (such as "reasonableness" or "best efforts") can be transposed into a program code. Converting those standards into a code means reducing them to the form and boundaries of the programming language. This reduction is complex and demanding, if it is possible at all. For the time being, the question of applying smart contracts as a complete replacement for traditional contracts is without a final answer. It is clear that traditional contracts cannot and should not be replaced overnight. There is no revolution in that sense. Changes must be made step by step. It is also necessary to devise the criteria for the identification of actors liable for legal compliance (to define the so-called "regulatory access point"). In addition, there are two possible suggestions regarding the types of liability in smart contract: strict liability for any fault in a code, and liability based on the reasonable-care standard. The former brings about legal clarity, but disincentivize the innovative potential of the Blockchain/smart contract technology; the latter may results in a vice versa outcome.

The main goal of Blockchain application in the context of smart contracts is to make the contractual relationship more efficient and economically viable, with fewer opportunities for contractual breach and subsequent disputes. With full respect towards national legislative efforts to define smart contracts, their legal conditions and consequences, the "state-of-the-art" approach is to allow for the smart contracts to reach maturity as a universal phenomenon. This process will sharpen its basic structure, clarify the most critical issues, and provide a catalogue of possible solution, the most important of which is liability.

References

Buterin, V. (2014). Ethereum White Paper: A Next Generation Smart Contract & Decentralized Application Platform, accessed 5. 03. 2020, https://blockchainlab.com/pdf/Ethereum_white_paper-a_next_generation_smart_contract_and_decentralized_application_platform-vitalik-buterin.pdf

Cvetković, P. (2020). Blokčejn kao pravni fenomen: uvodna razmatranja, Zbornik radova Pravnog fakulteta u Nišu, 87, 127-144.

Ducas, E., Wilner, A. (2017): The security on Financial Implications of Blockchain Technologies: Regulating Emerging Technologies in Canada, International Journal, No. 72/2017, 538-562.

European Commission (2018): Study on Blockchains: Legal, Governance and Interoperability aspects (SMART 2018/0038), report (28 February 2020), Luxemburg; accessed on 1. 05. 2020. https://ec.europa.eu/digital-single-market/en/news/study-blockchains-legal-governance-and-interoperability-aspects-smart-20180038

Grigg, I, (1996). The Ricardian contract, Systemics Ltd., Arkansas, accessed 2.06. 2020, https://www.iang.org/papers/digital_trading.html; http://iang.org/papers/ricardian_contract.html;

Kaulartz, M., & Heckmann, J. (2016). Smart Contracts-Anwendungen der Blockchain-Technologie. Computer und Recht, 32(9), 618

Lauslahti, K, Mattila,J & Seppälä, T (2017). Smart Contracts – How will Blockchain Technology Affect Contractual Practices? ETLA Reports No 68; https://pub.etla.fi/ETLA-Raportit-Reports-68.pdf, accessed 13. 04. 2019

Leiner, B.M.; Cerf, V.G.; Clark D.D.; Kahn, R.E.; Kleinrock, L.; Lynch, D.C.; Postel, J.; Roberts, L.G, Wolff, S. (1997), 'Brief History of the Internet' (Introduction), Internet Society, (published in 1997); https://www.internetsociety.org/internet/history-internet/brief-history-internet, accessed 1.3.2020.

Lessig, L, (2000). Code is Law, Harward Magazine, 2000; accessed 01. 07. 2020. https://davelevy.info/wiki/wp-content/uploads/2014/05/Code-is-Law.pdf

Sherborne, A. (2017). Blockchain, Smart Contracts and Lawyers; International Bar Association, 2017; https://www.ibanet.org/Document/Default.aspx?DocumentUid=17badeaa-072a-403b-b63c-8fbd985d198b , accessed 01. 08. 2019.

Szabo, N. (1996). Smart Contracts: Building Blocks for Digital Markets, accessed 1. 04. 2020. http://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/

CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/smart_contracts_2.html.

UK Government Office for Science (2016): Distributed Ledger Technology: beyond block chain. A report by the UK Government Chief Scientific Adviser, London, UK, 2016, accessed on 01. 08. 2019; https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/492972/gs-16-1-distributed-ledger-technology.pdf

Illustration of Merkle Tree: Göthberg, D. (2005): Hash Tree, English Wikipedia, released by D.G. as a public domain on 20 August 2005; retrieved on 01. 09. 2019 from https://en.wikipedia.org/wiki/Merkle_tree#/media/File:Hash_Tree.svg

Legislative acts

Malta Digital Innovation Authority Act C901 of 11 November 2018, Republic of Malta; accessed on 01. 09. 2019; http://justiceservices.gov.mt/DownloadDocument.aspx?app=lp&itemid=29080&l=1,

The Act amending Section 44-7003, Arizona revised Statutes; amending Title 44, Chapter 26, Arizona revised Statutes, by adding Article 5; relating to Electronic Transactions; State of Arizona, USA; https://legiscan.com/AZ/text/HB2417/id/1497439 , accessed on 01. 08. 2019.

Др Предраг Цветковић,
Редовни професор Правног факултета,
Универзитет у Нишу

ПРАВНА ОДГОВОРНОСТ У КОНТЕКСТУ ВЕЗЕ БЛОКЧЕЈН ТЕХНОЛОГИЈЕ И ПАМЕТНИХ УГОВОРА: уводна разматрања

Резиме

Блокчејн је појава која својом важношћу захтева правне одговоре на питања окренута његовом применом. Његова кључна карактеристика је да доноси промену парадигме у погледу улоге поверења уговарача: поверење у уговорног партнера замењује поверење у технологију. Пример наведене промене је концепт паметних уговора. Кључна карактеристика паметних уговора је да су само-имплементирајући. Ова карактеристика заснована је на томе да се обавезе страна извршавају кроз функционисање компјутерског програма (кода). Уговорне стране паметног уговора дефинишу права, обавезе, дужности и одговорности употребом програмског језика. У случају да наступи догађај који је предвиђен кодом (на пример, кршење уговора), код аутоматски извршава наредбу која је повезана са наведеним догађајем. Правно дејство паметних уговора не може да се негира искључиво с позивањем на чињеницу да сагласност страна о уговорним одредбама није дефинисана на начин како је то уобичајено у уговорном праву. Препреку за имплементацију и потпуно прихватање паметних уговора представља дискрепанција у развоју правних концепција одговорности и напретка технологије (при чему је потоња динамичнија). Отворено питање код паметних уговора је на који начин, и да ли је уопште могуће, обезбедити њихову флексибилност као елеменат готово свих контрактуалних инструмената. У погледу типова одговорности, постоје два приступа: принцип одговорности дизајнера кода за његово функционисање (па и функционисање паметног уговора који се тим кодом уређује) заснован на схватању његове обавезе као обавезе резултата и принцип одговорности заснован на природи обавезе дизајнера кода као обавезе средства (стандарди поступања у складу са најбољим напорима, разумности и слично). Први принцип стимулише правну сигурност, али дестимулише иновативни потенцијал блокчејн технологије (с обзиром на потенцијалне трошкове тог развоја). Други приступ даје подстицај за развој Блокчејн концепта, уз мањи степен правне одговорности дизајнера кода за штету причињену грешкама у коду (односно грешкама у функционоснају паметног уговора који регулише). Оптималан приступ правне заједнице је надгледање процеса сазревања паметног уговора као свеопштег феномена: очекивано је да ће то сазревање допринети дефинисању основне структуре описане технологије и обезбедити каталог одговора за решење кључних питања.

Кључне речи: *Блокчејн, "паметни" уговори, правна одговорност, технологија дистибуиране главне књиге, Рикардијански уговор, принцип "If this than that".*