INTERNATIONAL JOURNAL OF DIGITAL LAW – IJDL
ano 05 · n. 02 · maio/agosto 2024 – Publicação quadrimestral
DOI: 10.47975/digital.law.vol.5.n.2

05

ISSN 2675-7087

IJDL

International Journal of DIGITAL LAW

NUPED
Núcleo de Pesquisas em Políticas Públicas e Desenvolvimento Humano da PUCPR

FORUM

Licenciado sob uma Licença Creative Commons
Licensed under Creative Commons

International Journal of Digital Law – IJDL
ISSN 2675-7087
DOI: 10.47975/digital.law.vol.5.n.2.niebla

Smart contracts: the new method of interaction between the law and technology*

Contratos inteligentes: El nuevo método de interacción entre el derecho y la tecnología

Jesus Manuel Niebla Zatarain**
Universidad Autónoma de Sinaloa (Culiacán Rosales, Mexico)
j.niebla@uas.edu.mx
https://orcid.org/0000-0001-8460-4538

Paola Jackeline Ontiveros Vázquez***
Universidad Nacional Autónoma de Mexico (Ciudad de Mexico, Mexico)
paolaontiveros@hotmail.com
https://orcid.org/0000-0003-1460-7914

Recebido/Received: 03.09.2024/ September 9th, 2024
Aprovado/Approved: 14.10.2024/ October 10th, 2024

Abstract: Technology has reshaped the law. The Internet and derived technologies have led to the adaptation of traditional legal figures with the objective of bringing certainty to users and developers. A field that has been subject of constant technological development is contact law. This paper will address this scenario from the perspective of smart contracts, which allows not only a digital representation of

Como citar esse artigo/*How to cite this article*: NIEBLA ZATARAIN, Jesus Manuel; ONTIVEROS VÁZQUEZ, Paola Jackeline. Smart contracts: the new method of interaction between the law and technology. *International Journal of Digital Law*, Belo Horizonte, ano 5, n. 2, p. 103-123, maio/ago. 2024. DOI: 10.47975/digital.law.vol.5.n.2.niebla.

* This article is part of the project "Regulation of Digital Environments through Legal Reasoning Based on Artificial Intelligence" code CF-2023-G-772 sponsored by the National Council of Humanities Science and Technology of Mexico ("Regulación de Entornos Digitales a través derazonamiento legal basado en inteligencia artificial" claveCF-2023-G-772 del Consejo Nacionalde Humanidades, Ciencia y Tecnología (CONAHCYT) de México

** Professor and Reseacher at the Faculty of Law of Mazatlan of the Autonomous University of Sinaloa (Culiacán Rosales, Mexico). E-mail: j.niebla@uas.edu.mx.

*** Professor attached to the Postgraduate Law Program at the National Autonomous University of Mexico (Ciudad de Mexico, Mexico). E-mail: paolaontiveros@hotmail.com.

the obligations agreed by the parties, but also the capacity to solve discrepancies to ensure operation and the lawful fulfillment of its objective. Finally, this joint approach offers compatibility with transactions that take place in digital scenarios, contributing to a safer and law compliant cyberspace.

Keywords: Artificial intelligence; contracts; artificial legal reasoning; legal informatics; digital platforms.

Resumen: La tecnología ha remodelado la ley. Internet y las tecnologías derivadas han propiciado la adaptación de figuras jurídicas tradicionales con el objetivo de aportar certidumbre a usuarios y desarrolladores. Un campo que ha sido objeto de constante desarrollo tecnológico es el derecho de contactos. Este trabajo abordará este escenario desde la perspectiva de los contratos inteligentes, que permiten no solo una representación digital de las obligaciones acordadas por las partes, sino también la capacidad de resolver discrepancias para asegurar su operación y el debido complimiento de su objeto. Finalmente, este enfoque conjunto ofrece compatibilidad con transacciones que tienen lugar en escenarios digitales, contribuyendo a un ciberespacio más seguro y compatible con la ley.

Palabras clave: Inteligencia artificial; contratos; razonamiento jurídico artificial; informática jurídica; plataformas digitales.

---

1 Introduction

Technological development has traditionally impacted the legal field, whether through the regulation of a new scenario or by expanding already existing legal provisions. Nonetheless, this relation which has occurred in a traditional and escalated manner has been changed abruptly with the arrival of the cyberspace. This development expanded human interaction in an unforeseen volume and facilitated the migration of human activities to digital platforms through the use of the computational technologies, such as artificial intelligence. In this sense, a legal element that has been subject of technical approach are contracts, now referred as smart contracts. These agreements operate through platforms that represent the interests of the parties, guarantying the fulfilment of the contracted obligations. Within this new techno-legal paradigm, the position of the legal framework has shifted to a cooperative role, no longer being an independent subject that intervenes *ex post* but one embed to the technological element. However, these developments have recurrent errors derived of the incompatibility between natural and formal expressions of languages, which generates inconsistencies that impact the construction of the legal outcome. To solve this, a deterministic approach will be presented, which will also facilitate harmonization between different jurisdictions. This will enhance the legal representation process within the smart contract, providing legal certainty to the parties involve and facilitating its dissemination within the techno-legal market.

2 Smart contracts: making obligations "digitally" smart

Law is experimenting a transformation process as a direct result of the development of intelligent technology. This can be seen either through the expansion of regulatory frameworks to include virtual environments or by collaborating in the development of cognitive modules to allow law compliant operation of automated devices. Regarding the last point, the arrival of cyberspace and the dissemination of ubiquitous technology have incremented this approach developing new forms of legal interaction, this is the case of smart contracts. Nonetheless, there is a vast variety of definitions regarding this technology, one of the accepted ones states that "a smart contract is an agreement whose execution is both automatable and executable".[1] In this matter, Szabo (1996) defines these developments as "a set of promises, specified digitally, which include protocols in which the parties carry out a series of premises". Similarly, Alharby and van Moorsel state that smart contracts are "executable code that operates on blockchain to facilitate, execute and enforce the terms of an agreement. The main objective of smart contracts is, consequently, to automatically execute the terms of said agreement ".[2] Notably, these developments are considered not only as an extension of a traditional legal act in digital platforms, but as a new mechanism capable of delivering legal certainty to the parties under a technological scheme. In relation to its operation, it is important to note that although it is a largely automated approach, some parts may still require human intervention and control. In this sense, its operating model includes the legal application of rights and obligations as well as tamper-proof execution.[3] The implementation and automatic execution of the contract is based on legal knowledge translated into computational code.[4] Consequently, the execution of the content of the contract is not based on the human element, but on the representation of human interaction through computers, producing effects on both physical and digital environments. The operation of these developments can be seen in a variety of scenarios, for example, when a customer has acquired a particular type of bank financing, certain transactions will automatically occur if predetermined conditions are met. In this way, if the client defaults on the payments foreseen as part of their loan, the contract can trigger a particular type of consequence. As can be inferred, a smart contract seeks to replicate not only the regulatory framework applicable

---

1 MADIR, J. *Smart Contracts (How) Do They Fit Under Existing Legal Frameworks?*, 2018.

2 ALHARBY, M.; VAN MOORSEL. A. Blockchain-based smart contracts: A systematic mapping study, in International Conference on Cloud Computing, Big Data and Blockchain (ICCBB), 2018, p. 2 .*Conference on Cloud Computing, Big Data and Blockchain (ICCBB),* 2018, p. 2.

3 CLACK, C. D.; BAKSHI, V. A.; BRAINE, L. Smart contract templates: foundations, design landscape and research directions, *Cornell University,* 2017, p. 8.

4 T. HVITVED, "Contract formalisation and modular implementation of domain-specific languages", in *Doctoral dissertation, PhD thesis, Department of Computer Science, University of Copenhagen*, 2011 p. 4-5.

to a particular scenario, but also the interaction process inherent to the parties. Regarding its classification, Raskin (2016) establishes the existence of two types of smart contracts: rigid and weak. The first category encompasses those that have prohibitions, including revocation and modification of terms. On the other hand, weak smart contracts do not possess these characteristics. From a legal point of view, this classification can be seen as follows: if a court is able to effectively reverse[5] or alter the operation through a legal order, it is a weak smart contract. Conversely, if altering the contract to reverse its results proves too computationally costly for the court, then it is a strong smart contract. The complexity of these digital approaches is further intended to provide a method for resolving dispute.[6] Considering its configuration, the aforementioned author presents two approaches: traditional and non-traditional application methods.[7] The first classification refers to those contracts designed to resolve disputes through the implementation of elements used by legal courts. Non-traditional media target those that are technologically more complex and have little or no opportunity to perform differently from what was originally coded (they may still have software flaws, however). This differs from traditional legal positions as the proposed solution is reasoned by the cognitive module of the smart contract. The foregoing has led to situations in which a techno-legal opinion is required depending on the nature of the scenario.

2.1 Smart contracts: Generalities

Smart contracts operate through a logical representation of contract law and respond to the particularities of the interaction of the parties in a given scenario. In general, these devices operate under two minimum elements:[8] 1. Deliver a version equivalent to a contract concluded in the physical world, guaranteeing its security through cryptographic methods, and 2. These developments are compatible with automated technology. However, beyond the apparent novelty of this approach, it is important to note that it was first presented in 1994.[9] The adoption of these devices has been limited, among other factors, by the positions coming from traditional sectors, which are reluctant to accept them as representations of their physical counterparts, which, in many cases, is the result of interpretation problems. In this sense, when one of the parties presents their case before a Court, the legal operators

5 This term refers to breaking down the structure of a smart contract.
6 STARK, J. *Making sense of blockchain smart contract*, available at: https://www.coindesk.com/making-sense-smart-contracts.
7 STARK, J. *Making sense of blockchain smart contract*, available at: https://www.coindesk.com/making-sense-smart-contracts.
8 LADLEIF, J.; WESKE, M. *A Unifying Model of Legal Smart Contracts*, in *International Conference on Conceptual Modeling Springer*, 2019 p. 3-7.
9 SZABO, N. Formalizing and securing relationships on public networks. *First Monday*, 1997, p. 10.

base their decision exclusively on the operation of these devices. However, smart contracts (especially those considered rigid) are incompatible with this position due to their technological architecture. These devices operate through a self-executing protocol which, to be analysed, requires that they conclude their execution. It is important to note that this characteristic should not be understood as a negative element in the strict sense, since it fulfils the purpose of significantly reducing the possibility of presenting operational errors during the execution stage of these contracts. This ends up being an advantage over traditional legal agreements since the fulfilment of the conditions established in the contract.

To avoid the drawbacks of this approach, the adoption of contractware is recommended.[10] Naturally, both for the legal and the technological position, its performance and application will depend on the level of compatibility of said conditional structure.[11] Additionally, it is important to note that the judge may order the operational description on which the smart contract operates, the role delivered by the parties, as well as the conditions and their representation within the contract. This scenario led to the constant development of new collaborations within the technology sector, one of the most important being the adoption of blockchain. In the next section, this approach will be addressed.

3 Let's play it safe: Blockchain and smart contracts

One of the most important technical components of smart contracts is the use of blockchain technology. This digital approach is defined by Böhme et al as "a decentralized collection of data which is analysed by members of a point-to-point network. In this sense, Satoshi Nakamoto in 2010 He said possible about the blockchain "its design allows the development of a tremendous variety of transactions. Escrow transactions, linked contracts, third party arbitration, multiple signatures, etc."[12] Regarding its technical composition, blockchain is a database distributed over a network which keeps a record of all the transactions that take place within it. This database is replicated and shared by the participants and must communicate and deliver transactions between each of them in a secure way without the participation of third parties. Each block is linked to the previous one, thus resulting in a chain of blocks, thus generating a series or chain of transactions.

---

10 For further reading see: M. RASKIN, *The law and legality of smart contract*, in *Georgetown Law Technology Review 304*, 2017.

11 Dicha postura fue mencionada es propuesta en: J. MCCARTHY, *Recursive functions of symbolic expressions and their computation by machine*, in *Communications of the ACM 3.4*, 1960.

12 M. SWAN, *Blockchain: Blueprint for a new economy*, in *O'Reilly Media, Inc.* 2015, p. 10.

After a block is created and appended to said chain, the transactions contained in it cannot be modified. This guarantees the integrity of these operations and increases their efficiency, preventing the problem of double investment of resources.[13]

However, the implementation of blockchain delivers a new set of legal and governance challenges. From a law enforcement perspective, merging legal concepts and smart contracts present a range of conceptual and technical problems,[14] which must be approached from an equitable perspective for both the legal and technical sectors. A promising approach to solve this relies the adoption of deontic logic to properly represent legal logic expressions through computational code. This allows not only to properly represent contract law but also to include potential situations that may affect the execution of the obligation of the parties.

3.1 Blockchain and smart contracts: the relation grows

Without a doubt, one of the most influential developments in smart technology in the last years are smart contracts. Szabo defines them as “a set of promises, specified in digital form, including protocols within which the parties perform on these promises”. Another position defines them as “executable code that runs on the blockchain to facilitate, execute and enforce the terms of an agreement. The main aim of a smart contract is to automatically execute the terms of an agreement”.[15] Regarding its technical composition, these developments offer fees considerable lower than those provided by traditional systems that require a third party to enforce the terms contained on the agreement. Regardless its apparent novelty, the original conception of what it is called today smart contracts was first conceived in 1994,[16] but it was until blockchain technology emerged that the idea finally became a reality. Additionally, these developments proved compatible with electronic data interchange (EDI) formats, which have been used for several decades now to communicate digitally across supply chains.[17] A technical approach that has deeply impacted Contract Automation is the Ricardian Contract. “A Ricardian Contract can be defined as a single document that is a) a contract offered by an issuer to holders, b) for a valuable right held by holders, and managed by the issuer, c) easily readable by people [...], d) readable by programs [...], e) digitally signed, f) carries

[13] M. ALHARBY and A. VAN MOORSEL, *op. cit.* p. 6-7.

[14] G, GOVERNATORI, F, IDELBERGER, Z. MILOSEVIC, R. RIVERET, G. SARTOR, X. XU, *On legal contracts, imperative and declarative smart contracts, and blockchainsystems*, in *Artificial Intelligence and Law 26 (4)*, 2018, p. 395.

[15] M. ALHARBY and A. VAN MOORSEL, *op. cit.* p. 4.

[16] N. SZABO, *op. cit.* p. 11.

[17] To know more about EDIs, visit: What is EDI (Electronic Data Interchange)? *EDI Basics*, 2024. Available in: https://www.edibasics.com/what-is-edi/. Last accessed August 27th 2024.

the keys and server information, and g) allied with a unique and secure identifier".[18] This approach was the first to deliver a contract suitable of being understood by humans and computers. Consequently, it is not only legally enforceable but will also lend itself to analysis by and interaction with software.[19] This concept is easily extendible to cover Contracts other than token issues. Key features of the Ricardian Contract is the concept of a single document that holds both the Natural Language Contract and its logical description in a language capable of being interpreted by a computer. In section 5, potential issues on the automation of Contracts will be discussed. A Ricardian Contract needs an underlying platform capable of providing certain features and properties to such digital construct:

1. Immutable storage, as the platform needs to hold a reliable source of contract obligations and operations that affect such contract.
2. Strict relative ordering of events, as it is key that contract clauses are executed in the order stated.
3. Deterministic execution of programs, to make sure that given a particular contract and a set of events occurring in a particular order, the state and outcome of the contract should be the same.
4. Support digital signatures, to support authorship of the operations that affect the contract (such as issuing it, signing it, and operating on it).

In this regard, Blockchain is a suitable technology to automate contracts. This was the case since the dawn of this technology in Bitcoin.

3.1.1 A brief description of the technical nature of blockchain in smart contracts

The operational architecture of smart contracts presents two attributes:[20] 1) value and 2) state. The operational structure of these devices is structured on an *if – then* architecture. In this sense, these platforms function on previously established and agreed terms, which are later submitted to the blockchain network in the form of transactions. Each and every transaction is dependant to one or more secrets linked to the identities of the parties of the contract and all parties must recognize such links. Once the transactions are broadcasted via P2P network, they are analysed and confirmed by the miners and, eventually, placed on a particular

[18] I. GRIGG, *Financial cryptography in 7 layers*. In *International Conference on Financial Cryptography*, Springer, 2000, p. 332-348.

[19] I. GRIGG, *Financial cryptography in 7 layers*. In *International Conference on Financial Cryptography*, Springer, 2000, p. 332-348.

[20] S. WANG, OUYANG, L. YUAN, Y. NI, X. HAN, & F. Y. WANG, *Blockchain-enabled smart contracts: architecture, applications, and future trends*, in *IEEE Transactions on Systems, Man, and Cybernetics: Systems,* 49(11), 2019, p. 2266-2277.

block. The parties involved in the contract received the returned parameters, after this, users can invoke a contract by sending a transaction. The transaction is verified by miners who operate through the system's incentive mechanism. Particularly, after the miners receive the "contract creation" transaction or invoking transaction, they register contract or execute contract code. If the conditions are properly fulfilled, the response actions are executed. After an additional validation, this transaction is located into a new block that is added to the blockchain after the whole network has reached a consensus. So, basically, the only way that the smart contract state can change is as a consequence of one or more transactions. This means that smart contracts cannot change its state as effect of externalities. Examples of externalities desirable to affect the state of a contract include the passing of time, information from external sources (such as commodity or forex prices, occurrence of events) or changes of states in other smart contracts.

For a smart contract to interact with the reality, it requires a component that allows translating events in the real world to blockchain transactions that effect the smart contract. This component is referred to it as an *oracle*. There is a taxonomy of oracles,[21] but here there will be noted that oracles can be inbound (allowing a change in the smart contract state as consequence of an externality) or outbound (allowing the smart contract to trigger a real world action as a consequence of a change in the smart contract state).

The operational structure of smart contracts aims to replicate elements contained on their physical counter parts. Nonetheless, this is not always a welcomed feature, particularly for lawyers. For example, a smart contract that operates within parties that no longer wish to be remained legally bonded may still cause legal obligations. Naturally, the response proposed by developers is to generate a "termination button", which would operate if and only if certain requirements are established by the parties prior the beginning of its operation. What remain as a complex task yet to be solved, is how to deal with unforeseen events that occur and have a direct effect on the original object for which the contract was originally conceived.

3.1.2 Blockchain platforms used for smart contracts

Bitcoin, as defined in the original protocol, is a suitable and scalable platform for the negotiation, deployment and execution of private yet publicly auditable smart contracts. These characteristics make it an attractive option to complement smart contracts, allowing them to operate dynamically, efficiently and in a law compliant manner. Its scripting language allows for the implementation of state machines that

[21] See: S. VOSHMGIR. *What Is the Token Economy?*. In *O'Reilly Media, Incorporated.* 2019.

track the states of individual clauses within a contract, and the state of the contract at large. It allows the registration not only of the code of the smart contract, but also its related natural language legal counterpart.[22] This delivers a suitable feature already mentioned on this article: the compatibility between natural language and programming.

A disadvantage of Bitcoin as a component of smart contracts, is that the toolchain (defined as the set of tools that the developers of smart contracts require to program, interact and monitor) is not as mature as in other platforms. Many smart contracts dealing with tokens are developed on Ethereum. This platform operates as a transaction-based state machine: it starts with genesis states and, as it executes transactions it morphs into some final states.[23] Additionally it operates on accounts, which can be classified as follows: 1) externally owned accounts (EOA) and contract accounts. The differences between these two is that the first classification operates on private keys without code associated with them, whereas the second one operates on contract code associated with associated code.

Transactions can only be initiated through EOAs and it can include Ether (the cyber coin of Ethereum or binary data (payload). After this the management process begin until miners finally verify the proper structure of the contract. Relevantly, all interactions with the Ethereum public blockchains are subject to fees, which are covered on gas.[24] This, with the intention of avoiding unnecessary network abuse and other potential problems. The major disadvantage of Ethereum as a platform for smart contracts are scalability issues proper of its architecture and the fact that once, the controllers of such platform performed a rollback of a transaction (the DAO incident).[25] Another platform to develop smart contracts is Hyperledger Fabric. Hosted by the Linux Foundation, it is not a blockchain developing platform but a Distributed Ledger Technology. It operates on a private approach, which restricts access to it only to membership holders. This network is developed and maintained by contributions made by peers belonging to different groups within this organization. In this sense, these peers are hosts for ledgers and chaincodes (smart contracts). The ledger is structured as a sequence, unmodifiable record of transactions and/or state of transitions. Every transaction is turned into a set of asset-key value parties that are committed to the ledger as they create, update or delete. The operative approach of the Hyperledger Fabric is composed of three phases:

---

22 WRIGHT, C.S. Turing Complete Bitcoin Script White Paper. *SSRN* n. 3160279.

23 WANG, S.; OUYANG, L.; YUAN, Y.; NI, X.; HAN; WANG, F. Y, *op. cit.* p. 2268.

24 WOOD, G. Ethereum: a secure decentralised generalised transaction ledger Shangai Version. 47e97f5 – 2024-08-26. *Ethereum & Parity*, 2018, available at: https://ethereum.github.io/yellowpaper/paper.pdf, last access on the 27th of August 2024.

25 COPPOLA, F. *Ethereum's DAO Hacking Shows That Coders Are Not Infallible*, 2016.

1. Proposal: In this phase, an application sends a transaction request to other peers. This is basically the request to read and/or write on a particular ledger. It is here where the running of the chain code will be executed.
2. Packaging: Here, the identity of the enforcer is confirmed through a process of signature analysis. After the previous is confirmed, it presents a proposal to modify or update the ledger. The ordering service sorts the transactions received from the network and packages batches of transactions into a block ready for distribution back to all peers connected to it.
3. Validation: The peers connected to the ordering service validate every transaction inside a specific block to confirm whether it has been properly endorsed by the validation requirements established by the organization's policy. After this process, every peer adds the block to the chain, updating the entire structure.

Some disadvantages of using Hyperledger as a platform for smart contracts is the lack of external auditability and the lack of a single source of truth across different implementations. The most evident difference between these developing platforms is their nature: Bitcoin and Ethereum are public blockchain platforms, while a Hyperledger Fabric implementation is private in nature where only a set of previously accepted users can participate. Another difference relies on the fact that in Hyperledger Fabric there is no cyber currency. As part of this, it only defines a set of assets, which are presented as key–value pairs, and provides the functions for operating on the assets and changing their states. Lastly, in relation to contract execution, unlike Ethereum, it is hosted by peer nodes. After a transaction is created, it is only executed and signed by specific peers. Overall, regardless the apparent opposition of these developing platforms, it is important to mention that collaboration between these two developing platforms is becoming more common.

4 Code is (contractual) law?

To this point, this paper has argued that there is yet no definitive method to properly translate natural language (in which the law is expressed) to programming code. This may lead to unforeseen yet legally important situations that may jeopardize the positions of the parties involved in the contract. To properly understand this scenario, one of the most important positions is presented by Larry Lessig in his "code is law approach", This allows technology and the law to be suitably combined by stating that technological artefacts can be embedded with values that constrain the volume of actions performable on them.[26] Nonetheless, there is a large

[26] LESSIG, L. Code is law. *Harvard Magazine*, 2000.

discussion of how this should be implemented.[27] Largely, software developers may implement one of two approaches: they either mirror a section of the law, which in turn may lead to an unnecessary demand of technical resources. The second, to design the operational after a particular legal provision, which in contrast may lead to over simplistic representations of the law. Inevitably, this leads us to Blockchain technology and smart contracts. Unlike other areas of law, traditional contracts are based upon of the convergence of the will of the parties to obtain the realization of a given object.

Smart contracts reduce traditional contractual relations to code-whereby clauses that are automatically enforced after pre-programmed conditions are met. To ensure security, these developments operate through a decentralized storage transaction approach, which decreases the chances of data corruption or lost. The notion of bounding the wills of the parties through technology has been subject of research and development for the digital industry long before the arrival of what we now call smart contracts. Consequently, this technology should be addressed as an evolution of the interaction between the law and technology instead of a new contribution. It is because of their dynamic nature that the proposal to provide them with the capacity of properly represent the law through computational code is a necessary feature.

4.1 How code is (contract) law

As mentioned in upper lines, computational code is suitable of representing legal provisions in technological devices, such as smart contracts. These are key elements that, if properly programmed, would allow these devices to deliver lawful compliance, based on a particular jurisdiction, according to the characteristics of the scenario. In this paper, this representation will be addressed in the following section from the perspective of the US and Mexican contract law in relation to sales contract. The Mexican Federal Civil Code states in its article 2248 that: “there will be a sales contract when one of the parties commits to transfer the property or a particular right of a thing to another, whom in turn commits to pay a price certain and in money.”

This provision can be represented in legal logical terms as:

a = seller
b = buyer
c = good

[27] WU, T. When code isn’t law. *Virginia Law Review*, 2003

```
if a offers_goodproperty_in_money c
    b (agrees=c)
    b (acquires_goodproperty=c)
    else
    b not(aquires_goodproperty=c)
End if
```

This representation contains three elements presented on the Mexican Federal Civil Code: the seller, the buyer and the thing object of the contract. It establishes that the seller offers a thing in a certain priced that must be agreed by the buyer. Once this has been met, the contract executes its effects.

From the perspective of the United States Jurisdiction, the Uniform Commercial Code states on its article 2 section 106: "Contract for sale" includes both a present sale of goods and a contract to sell goods at a future time. A "sale" consists in the passing of title from the seller to the buyer for a price. Due to its legal structure, it can be logically expressed in the same way as its Mexican counter part. Notably, this results compatible with The New Agreement between the United States of America, the United Mexican States, and Canada (USMCA).[28]

```
a = seller
b = buyer
c = good
if a offers_goodproperty_in_money c
    b (agrees=c)
    b (acquires_goodproperty=c)
    else
    b not(aquires_goodproperty=c)
End if
```

Notably, both legal logic expressions hold a notable degree of similarity, which allows them to be represented in practically identical terms This is a relevant feature in terms of software development and certainty for the parties, regardless of the jurisdiction they are from.

A technical expression compatible with these legal provisions is presented in the following lines:

```
/******* Payment state machine ***********/
Var contract = <natural language contract>
Var buyer = <buyer identity>
Var seller = <seller identity>
```

[28] United States Trade Representative, Office of, "*Agreement between the United States of America, the United Mexican States, and Canada 12/13/19 Text*", available at: https://ustr.gov/trade-agreements/free-trade-agreements/united-states-mexico-canada-agreement/agreement-between

```
Var goods = [ <object description 1>, <object description 2> ]
Var total_price = <total price>
Var escrow = <nul>
Var delivery_time = <time of delivery>
Var contract_state = <init> //init, partially accepted, fully accepted, done
Var payment_state = <none> //none, init, buyer_to_escrow, escrow_to_seller, escrow_to_buyer, done)
Var buyer_signature = <buyer signature>
Var seller_signature = <seller signature>
Var evidence_tracking_number = <nul>
/******* Sales contract expression ***********/
If (buyer_signature XOR seller_signature) contract_state = partially_accepted
If(buyer_signature AND seller_signature) {
contract_state = fully_accepted
payment_state = init
If (payment_state == done) contract_state = done.
/******* Payment state machine, executed in case ***********/
input: buyer_payment, seller_tracking_number, buyer_accepts_goods, buyer_rejects_goods
output: send_goods_command, return_payment, deliver_payment
if (payment_state == init) SEND payment_instructions TO buyer
ON buyer_payment:
Escrow = buyer_payment;
SEND send_goods_command TO seller
ON tracking_number:
Evidence_tracking_number = tracking_number
ON buyer_accepts_goods:
SEND escrow TO seller
ON buyer_rejects_goods:
SEND escrow TO buyer
```

4.2 Traditional contract law and smart contracts: a new boundary?

To this point it has been shown that the increasing adoption of smart contracts generates concerns to the traditional legal sector. Besides an adequate representation of legal terms, transjurisdictional scenarios have been raised serious concerns. In relation to this, to Savelyev (2017), the main problem relies in the technical approach implemented to develop smart contracts, which lacks a proper degree of participation from legal experts. This leads to the development of platforms with a limited version of the law thus, compromising legal certainty. Additionally, some of the following issues are presented in projects where and leading to the following issues:[29]

- Smart contracts do not generate legal obligations
- Vitiated consent cannot be plead in smart contracts
- Smart contracts are egalitarian platforms
- The duality of smart contracts: potential illegal uses

In the following section the adoption of smart contracts will be addressed from the perspective of international jurisdictions.

5 Smart contracts and international legal frameworks: an on-going relation

Smart contracts have raised the attention of the legal community since its very first conception back in 1996. Many positions have presented both, positive and negative aspects related to the potential consequences of this technology. However, it was during the 2016 attack when this technology reached its peak when a hacker stole 3.6 million Ether (the currency used on the Ethereum platform) from the Ethereum's Decentralised Autonomous Organisation (DAO). Shortly after this was reported, a statement was released claiming that the amount would be recovered thus, minimizing the risk for investors. Nonetheless, this led to questions about the composition of smart contract since, as it was stated by the cracker and later confirmed by DAO forensic experts, this was the result of self-executing transactions and it was not committed through traditional illegal actions.[30]

Regardless of this, the implementation of Blockchain has expanded in many markets. For example, in the US it is expected that by 2025 such growth will be worth around US$41-US$60bn. Parallel to this, it is also expected the arrival of new legislations at local and federal levels, which will have an impact on this market bringing certainty and increasing its adoption rates. In relation to the legal aspect, one of the most relevant provisions was presented in the state of Wyoming in 2019. This contained 13 Blockchain-enabling laws that provided property rights to users and creators and offered regulatory relief (Spindler 2019).[31] Arizona joined the regulatory effort by taking some the initiatives presented by Wyoming and keeping close collaboration with the technologic sector. Nevertheless, there are very few

---

[29] DUGGAL, P. Blockchain Contracts & Cyberlaw. *New York,* 2015.

[30] STEEMIT, *An Open Letter to the DAO and the Ethereum Community*, available at: https://steemit.com/ethereum/@chris4210/an-open-letter-to-the-dao- and-the-ethereum-community.

[31] C. LONG, *What Do Wyoming's 13 New Blockchain Laws Mean?*, available at: www.forbes.com/sites/caitlinlong/2019/03/04/what-do-wyomings-new-blockchain-laws-mean.

provisions directly related to smart contracts. This, since legislation perceives them merely as an extension of Blockchain development.

In relation to the European Union, there are two legal measurements currently being discussed, first, the Directive 2000/31/EC on e-commerce and second, the Consumer Rights Directive 2011/83/EU. If these two are passed, they will deliver provisions related to the formation of contracts on the Internet. For example, they would require pre-contractual obligations for a trader in e-commerce consumer contracts in the form of informing the consumer about relevant facts, which could be interpreted as also containing certain information about security vulnerabilities of smart contracts. In addition to this, the General Data Protection Regulation (GDPR) should also have a relevant role in the design and implementation of smart contracts. Latin American jurisdictions are also in the process of adapting their legal framework to the arrival of this technology. In the particular case of Mexico, the development of the Law to Regulate Institutions of Finance Technology provides relevant starting point towards the lawful and certain use of this type of technology. Regardless the fact smart contracts are not expressly contained on this law, they fall within the figure of "Novelty Models". These are required to establish all the necessary arrangements to facilitate the interaction between users and technological developments. In other words, authorities are allowed to keep a close contact with the organizations developing this technology to avoid the creation of unlawful implementations. In relation to Blockchain, the term is not properly mentioned on the law. However, the section related to "digital assets" is configured in such a manner that can be addressed as a manifestation of this technology. Digital technology presents a constant challenge for international jurisdictions. In the particular case of smart contracts, legal frameworks around the world are currently on the process of designing a legal figure suitable to regulate the activities derived from the adoption of these developments. Nonetheless, the development of legal figures to regulate smart contracts should be performed taking in consideration the principle of legal harmonisation. This will increase the level of success of law enforcement strategies, delivering legal protection to law abiding users regardless of their geographical location.

5.1 Smart contracts, blockchain and international legal framework

One of the areas that is expected to receive the impact of smart contract technology is, without a doubt, international trade. This, as result of some of the following reasons:[32] international companies are increasingly accepting virtual

[32] BALLY, G. *Cryptocurrencies accepted by Switzerland's biggest online retailer*, available in https://www.swissinfo.ch/eng/bitcoin-or-cash_cryptocurrencies-accepted-by-switzerland-s-biggest-online-retailer/44835480

currencies as a form of payment, blockchain offers not only a significantly smaller fees but also safer transactions. This has led us to a new scenario where decentralized relations and application present themselves as new challenges for the legal framework. In fact, this has already taking place: Ethereum provides an approach where third parties are no longer required and (smart) contracts can produce their effects in a more efficient manner. Speaking from a purely legal perspective, there is no global consensus of how smart contracts should be regulated on an international level, however, it seems to be a general consensus on three main elements[33] 1. Payment/exchange/currency tokens (virtual currencies or cryptocurrencies), 2. Investment tokens, and 3. Utility tokens. Within this scenario, this paper aims to fulfil this gap by allowing a representation that results compatible with the jurisdictions presented on the previous section. To achieve this, through the use of a logical description, legally relevant terms will be translated into a computational from that suits legal requirements from the positions of the parties.

6 Code is not perfect

The technical nature of the Blockchain and its relation with Smart Contracts presents challenges on different steps, which include development, deployment, updates, arbitration and interaction with reality of such Smart Contracts. This section will overview such challenges.

6.1 Potential issues in the lifecycle of a smart contract

The following are the issues that have been identified per type within the lifecycle of a smart contract.

A. Defects of origin. This type of issues appears when the law or the source legal contract has a defect, which can be: contradiction, de-harmonization of the contract with the body of law and lack of consideration to different scenarios, amongst others. The consequences of defects are the same for any traditional Legal Contract in this situation (mainly high contractual risks, litigation costs, etc).
B. Defects in the Logical Abstraction process: Wrongly identifying actionable clauses, sub-optimally selecting actionable clauses for automation, errors in the logic abstraction of clauses, lack of consideration of the properties and limitations of the underlying technology (i.e. oracles).
C. Defects in the Smart Contract. Those are the defects not found in the validation phases.

[33] EUROPEAN COMISSION, *Legal and regulatory framework of blockchains and smart contracts thematic report*, available at: EUBlockchain, 2019.

D. Defects in the Smart Contract's dependencies. Examples of these are defects in external libraries, compilers, debuggers or applications of the toolchain.
E. Defects in the Smart Contract's Platform (SCP). Examples of these are problems in the deployment of the contracts or problems in the upgrading of the contracts.
F. Defects in the inbound or outbound oracles. Examples of these are downtime, erroneously serving information, wrongly executing actions.
G. Defects in the Oracles' dependencies. These are mainly due to downtime of the underlying mechanisms, such as payment service suppliers.

Consequences of critical severity are a potential outcome of the defects B, C, D, E, F and G, because they can cause the Smart Contract system to behave differently than the agreed legal contract, thus, increasing contractual risks and potentially, litigation costs. There could also be minor severity consequences in defects D, E, F and G, which could cause a delay in the actions as consequences of the Smart Contract's logic.

6.2 Proposal for the development of a technical process for smart contracts

The roles involved in the proposed development process of a Smart Contract are: Legal Contract Writer (LCW), Smart Contract Developer (SCD), Logical Abstraction Technical Lawyer (LA), Smart Contract Tester (SCT), Smart Contract Platform (SCP) and Oracles (O).

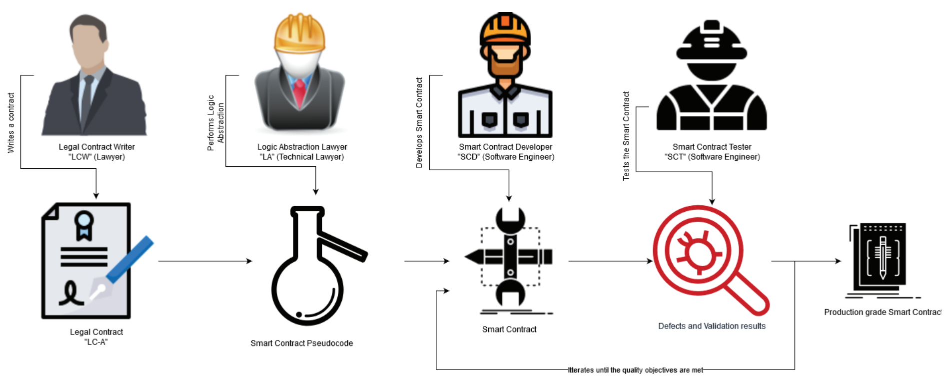

Figure 1: describes the Smart Contract development process.

The development process of a Smart Contract starts with a Legal Contract. Assuming that there is already a natural language, jurisdictionally bound Legal Contract "LC-A", it will be the trigger for the process of development of a Smart Contract. It is important that this process starts with the definition of the Legal

Contract, because it can use the standard mechanisms in place for validation and harmonization processes (which could include peer reviews), to reduce the risks associated with contracting. Once the legal contract "LC-A" is developed, a process of logic abstraction is started. The purpose of this process is to determine which clauses are declarative and which are actionable, then select which actionable clauses will be automated. Once the particular subset of actionable clauses subject to automation are identified, the LA role abstracts the logic on each of them. This is a critical part of the Smart Contract development. Since all the automatable actions are identified, selected and implemented here, careful consideration must be taken to ensure that harmonization with existing laws and jurisprudence, in order to reduce the contractual risks. In this process, it is also imperative that the LA role takes into consideration the limitations of the underlying technology *as a whole*, not just the blockchain part, but also its interactions with external libraries, oracles, action mechanisms, and inputs of information. Even though careful consideration is required, it is possible that errors are injected in the logic abstraction of the Legal Contract. The outcome of the Logic Abstraction process is a pseudocode of the Smart Contract, which holds the business rules to be implemented as a Smart Contract. The next step of this to actually write the Smart Contract. This process is executed by the SCD role and takes the Smart Contract pseudocode generated by the LA and implements it. To take a pseudocode and implement it as an executable code, there are different cognitive and technical decisions to be defined such as:

- Which data types should be used?
- How to properly organize the data structure of the Smart Contract?
- Which programming language (in case that options are available) should be used?
- Does the Smart Contract require external libraries?

It is important to mention that in Software Development, error injection is unavoidable and there is no way to guarantee a bug-free software (when not using formal methods). The outcome of this process is an executable code of the Smart Contract, which is tied to a particular platform and corresponds to the pseudocode generated by LA. The next step in the process is the validation of the Smart Contract. This step is performed by the SCT role. Although it is possible that both the SCD and SCT roles lay within the same person, it is highly recommended that the SCT's activities in this step is performed by a different person. The activities of this step is to formally validate the Smart Contract, taking as its specification the pseudocode. This activity involves the creation of a testing plan, test cases and validation reports. It is imperative to achieve 100% code coverage during the validation phase, to ensure that even the more conspicuous situations are validated. The outcome of this phase is a validation report detailing the functional and non-functional characteristics of the performance of the Smart Contract, plus the set of defects found. It is important to note that a lack of defects does not mean that the Smart Contract code is error free, and more often than not, it means that the validation phase was not as stringent as required. This may also contribute to properly detect the lack of compatibility between the legal logic expression of the law and its computational counterpart.

The Smart Contract development and validation phases should iterate, so the SCD role can fix the defects found by the SCT. Once a quality objective has been reached, the Smart Contract code "SC" is ready for production. However, as previously stated, the Smart Contract is self-contained. For it to be truly an automation of its Legal Counterpart, it must interact with reality.

These interactions might include:

1. Interaction with the parties of contract, to assert their will on different parts of the process.
2. Interactions with external sources of information, for instance, to receive information on event referenced in the smart contract.
3. Interactions with external actuators, for instance, to trigger a payment, a delivery of a good or service, to grant access to an intelligent lock, etcetera.
4. Interaction with an external arbiter.
5. Interactions with external authorities (e.g. Tax Institutions, AML Institutions).

In order to connect the "SC" and make it fully functional, such "SC" must be connected to an existing platform "SCP" that allows such connections and a new validation phase to begin: Integration Testing. Such smart contract platform "SCP" can be in house or an external service provided by a third party. The Integration Testing and System Validation must occur with SCD, SCT and SCP, across several iterations, to fix any defect found on the Smart Contract and/or the oracles. Once the system validation is completed, the Smart Contract is fully functional and can be deployed.

7 Conclusions

Smart contracts offer not only the capacity to operate on digital platforms but are also developments capable of adapting their behaviour based on cognitive features. This, by integrating the developing and legal process through an approach that substantially reduces the chances of presenting technical errors. The impact of this is both, legal and technological sound. In the first case, it brings legal certainty to the parties involved, reducing potential litigations due to the mistranslation of the legal clauses. In the second case, it increases the impact on the market by making smart contracts a reliable product, suitable of delivering legal compatibility. Overall, the presented process delivers a novel process of reasoning without reducing technical efficiency, allowing compatibility with digital dynamic scenarios. Finally, this contribution aims to encourage further research into the development legal technological solutions for digital scenarios and to provide compatibility among different jurisdictions.

References

ALHARBY, M.; VAN MOORSEL, A. *Blockchain-based smart contracts: A systematic mapping study*, in *International Conference on Cloud Computing, Big Data and Blockchain (ICCBB),* 2018, p. 2.*Conference on Cloud Computing, Big Data and Blockchain (ICCBB),* 2018.

BALLY, *Cryptocurrencies accepted by Switzerland's biggest online retailer*, available in https://www.swissinfo.ch/eng/bitcoin-or-cash_cryptocurrencies-accepted-by-switzerland-s-biggest-online-retailer/44835480

CLACK, C. D.; BAKSHI, V. A.; BRAINE, L. *Smart contract templates: foundations, design landscape and research directions*, in *Cornell University,* 2017.

COPPOLA, F. *Ethereum's DAO Hacking Shows That Coders Are Not Infallible*, 2016.

DUGGAL, P*. Blockchain Contracts & Cyberlaw*, in *New York,* 2015.

EUROPEAN COMISSION, *Legal and regulatory framework of blockchains and smart contracts thematic report*, in EUBlockchain, 2019.

GOVERNATORI, G.; IDELBERGER, F.; MILOSEVIC, Z.; RIVERET, R.; SARTOR, G.; XU, X. On legal contracts, imperative and declarative smart contracts, and blockchainsystems, in *Artificial Intelligence and Law 26 (4),* 2018.

HVITVED, T. "Contract formalisation and modular implementation of domain-specific languages", in *Doctoral dissertation, PhD thesis, Department of Computer Science, University of Copenhagen*, 2011.

LADLEIF, J.; WESKE, M. *A Unifying Model of Legal Smart Contracts*, in *International Conference on Conceptual Modeling Springer,* 2019.

LESSIG, L. *Code is law*, in *Harvard Magazine*, 2000.

LONG, C. *What Do Wyoming's 13 New Blockchain Laws Mean?,* available in: www.forbes.com/sites/caitlinlong/2019/03/04/what-do-wyomings-new-blockchain-laws-mean.

MADIR, J. *Smart Contracts (How) Do They Fit Under Existing Legal Frameworks?,* 2018.

STARK, J. *Making sense of blockchain smart contract*, in *coindesk,* https://www.coindesk.com/making-sense-smart-contracts.

STEEMIT, *An Open Letter to the DAO and the Ethereum Community*, in https://steemit.com/ethereum/@chris4210/an-open-letter-to-the-dao-and-the-ethereum-community.

SWAN, M. *Blockchain: Blueprint for a new economy*, in *O'Reilly Media, Inc.* 2015.

SZABO, N. *Formalizing and securing relationships on public networks*, in *First Monday,* 1997.

United States Trade Representative, Office of, "*Agreement between the United States of America, the United Mexican States, and Canada 12/13/19 Text*", in https://ustr.gov/trade-agreements/free-trade-agreements/united-states-mexico-canada-agreement/agreement-between

VOSHMGIR, S. What Is the Token Economy? In *O'Reilly Media, Incorporated.* 2019.

WANG, S.; OUYANG, L.; YUAN, Y; NI, X; HAN; WANG, F. Y. Blockchain-enabled smart contracts: architecture, applications, and future trends, in *IEEE Transactions on Systems, Man, and Cybernetics: Systems,* 49(11), 2019.

WOOD, G. Ethereum: A Secure Decentralised Generalised Transaction Ledger Shanghai Version 47e97f5 – 2024-08-26. *Ethereum & Parity,* 2018, available at: https://ethereum.github.io/yellowpaper/paper.pdf, last access on the 27th of August 2024.

WRIGHT, C.S. *Turing Complete Bitcoin Script White Paper*, in SSRN 3160279.

WU, T. When code isn't law. *Virginia Law Review*, 2003.

Informação bibliográfica deste texto, conforme a NBR 6023:2018 da Associação Brasileira de Normas Técnicas (ABNT):

NIEBLA ZATARAIN, Jesus Manuel; ONTIVEROS VÁZQUEZ, Paola Jackeline. Smart contracts: the new method of interaction between the law and technology. *International Journal of Digital Law*, Belo Horizonte, ano 5, n. 2, p. 103-123, maio/ago. 2024. DOI: 10.47975/digital.law.vol.5.n.2.niebla.

Informações adicionais

Additional information

Editores responsáveis	
Editor-Chefe	Emerson Gabardo
Editor-Adjunto	Lucas Bossoni Saikali