Universitat d'Alacant
Universidad de Alicante

Facultat de Dret
Facultad de Derecho

SMART CONTRACTS FROM A LEGAL PERSPECTIVE

FACULTY OF LAW

DEGREE IN LAW

FINAL DEGREE WORK

ACADEMIC COURSE [2017-2018]

AUTHOR:

TANASH UTAMCHANDANI TULSIDAS

ACADEMIC TUTOR:

DR. D. AURELIO LÓPEZ-TARRUELLA MARTÍNEZ

CONTENT

ABSTRACT

Smart contracts are a new technology emerging with force due to the opportunities and benefits it can offer. The main innovation is that the terms parties agree to regulate their relations are executed automatically by a computer program. The intention of this Paper is to examine if the Spanish Contract Law is prepared for smart contracts or its modification is needed. First, it is necessary to understand the blockchain technology it can deploy on and the features of smart contracts. Then, under a legal perspective, it analyses if they comply with the essential contract elements -consent, object and cause- and main concerns during its existence regarding formation, performance and breach. Finally, it concludes by recognizing that smart contracts go along with existing Contract Law principles, offers some remedies and encourage legislators and jurists not to ignore these contracts for legal security.

KEYWORDS

Smart contracts. Blockchain. Contract Law. Certainty. Autonomous. Enforceable.

I. INTRODUCTION

Smart contracts are basically computer programs that automatically execute the terms parties have agreed on to regulate their relations. The idea is that the agreement is self-enforced, making its modification very difficult so that it ensures the performance. If there is a conflict between the parties, the aggrieved one will go to court after an improper compliance or unjust enrichment because the smart contract would have already been executed or in the process of execution. This creates more certainty to the courts because the intent must be clearly shown in the smart contract.

This idea already existed – i.e. vending machines-, but it is in the recent years where the blockchain pretends to revolutionize the way we act using decentralized consensus. One of its most promising implementations is smart contracts, already being used[1]. However, this new technology arouses many doubts from a legal perspective and there is no specific mention of these contracts in the law, but clearly it must face it. This is where this Article takes place.

The objective here is to, once gathered the basic knowledge associated to this new phenomenon, answer if smart contracts are prepared to deal with the Spanish legal system and see if any change is needed in Contractual Law to facilitate its use and guarantee its legal effectiveness. Also, relevant questions like is it legally a contract and will it take the place of traditional contracts, will be answered.

This Paper is divided into three sections focused on different aspects of smart contracts:

Section II begins defining the blockchain technology, how it works, and its main characteristics aimed for a basic understanding of this complex matter and from a non-

[1] Vega, G. (2018). *Santander, BBVA, Sabadell, Bankia, Iberdrola, Gas Natural y Cepsa crean la mayor 'blockchain' de España*. [online] EL PAÍS RETINA. Available at: https://retina.elpais.com/retina/2017/05/30/tendencias/1496145136_731555.html [Accessed 6 Apr. 2018]. Red Lyra, as the biggest Spanish blockchain network, pretends to develop new systems so that any person or company can securely digitally identify themselves. In it composed by big companies like Observatorio Comillas ICADE-Everis, Banco Santander, Banco Sabadell, BBVA, Bankia, Cajamar, BME, Correos, Scytl, Everis, Grant Thornton, Garrigues, Roca Junyent, Iberdrola, Endesa, Gas Natural Fenosa, Ejaso, Notarnet, Cepsa o Wordline.

legal perspective. Out of the multiple applications it offers, smart contracts can be deployed on this ledger and be benefited by it. Next, we will try to comprehend the meaning of smart contracts, how the advantages it offers revolutionizes a traditional contract by reducing the risk and associated costs and offering certainty, as well as uses in the real world.

In section III there is a turn in perspective to be completely legal. Smart contracts are questioned not as a computer program, but as proper contracts. For that, the essential elements -consent, object and cause- and the form will be analysed.

Finally, in Section IV, assuming smart contracts in a legal sense, they will be examined in three different phases: 1) Formation and perfection; 2) Performance and modification; and 3) Breach and remedies. The purpose here is to signalize most problematic issues that these contracts do not share with conventional ones.

II. UNDERSTANDING BLOCKCHAIN AND SMART CONTRACTS

1. BLOCKCHAIN AND HOW IT WORKS

A published article[2] by the name of Satoshi Nakamoto in 2008 introduces the blockchain technology. Initially, it was created to register records and process bitcoins through online payments among parties without the intervention of a financial institution. A revolution as it conforms to be an alternative payment system without intermediaries.

When we talk about blockchain, we frequently associate it with bitcoin, and vice versa, but they are not synonyms. Although, bitcoin is one of the multiple possible applications for the blockchain technology and the most traded one.

1.1. What is blockchain?

It is undoubtedly an ingenious invention that has caught the attention of investors, companies, public authorities and media. The term *blockchain* has been used by different agents in various ways that can be quite confusing. Sometimes they are referred to as the "bitcoin blockchain" (or another virtual currency) or "the smart contract blockchain". In either case it can be defined as **a distributed database of organised economical transactions or everything of value in the digital world**. The data it contains is not meant to be modified between participating parties and does not require the intervention of a third party. In technical words, a blockchain is "*a distributed peer-to-peer*[3] *system of ledgers that utilizes a software (...), which negotiates the informational content of blocks of data together with cryptographic and security technologies*"[4].

The idea is quite simple, but the complexity arises when all the concerning elements must be considered to create this technology and by its multiple applications. This paper

[2] Nakamoto, S. (2008). *Bitcoin: A Peer-to-Peer Electronic Cash System*. [online] pp.1-2. Available at: https://bitcoin.org/bitcoin.pdf [Accessed 4 Feb. 2018].
[3] Peer-to-peer (P2P) is the way of communication data used in blockchain because it is decentralised, and each node keeps a copy of the ledger. It is also possible to talk about *white-listed* servers that operates the same why but restricted to certain nodes.
[4] Drescher, D. (2017). *Blockchain basics - a non-technical introduction in 25 steps*. p.35.

pretends to cover only one specific purpose: smart contracts. However, there are many concrete possibilities that have caught attention such as currencies, digital content, patents, e-voting and supply chains, among others. Also, like all technologies, it includes specific concepts that are not commonly used in business practices or legal language and can complicate its understanding.

1.2. What are the types of blockchains?

Blockchains can be categorised in two different ways[5]. First of all, a blockchain can be:

- **Public**: in this case anyone with an Internet connection and correct software can access the blockchain and read and add information. It is ideal for cryptocurrencies, such as bitcoin, ethereum, litecoin… so that any person is encouraged to access and operate with these values.
- **Private**: on the contrary, here the access and the consensus is controlled by determined participants. In this scenario the participants belong to an organization or group of organizations, for example, between a holding company and its subsidiaries.

The second categorization goes along with the previous one and attends to the requirements of the users to be authorised to participate in the network[6]:

- **Permissionless**: every user in the network can participate in the verification process following a determined consensus procedure, needn't of authorization. For example, bankchain (in a private blockchain) and bitcoin (public blockchain).
- **Permissioned**: only determined users are allowed to verify the data, check or add information to the ledger. This would be the case of hyperledger in a public blockchain. Also, the government, banks or public institutions could be permitted to intervene.

[5] Tasca, P., Aste, T., Pelizzon, L. and Perony, N. (2016). *Banking beyond banks and money: a guide to banking services in the twenty-first century*. Zurich: Springer, p.244.
[6] Mukhopadhyay, M. (2018). *Ethereum Smart Contract Development*. Birmingham: Packt Publishing, p.222.

The relation between this double classification can be summed up in Table 1:

TABLE 1: BLOCKCHAIN DOUBLE CLASSIFICATION[7]

	Permissionless (no restrictions on processors)	**Permissioned** (transaction processing performed by predefined users)
Public (no restrictions on reading blockchain data)	Every user can read transaction data. Every user can validate transactions in blocks. e.g. **Bitcoin / Ethereum**	Every user can read transaction data. But only predefined users can validate transactions. e.g. an Hyperledger Fabric instance on which transaction data is made publicly visible.
Private (direct access to blockchain data is limited to predefined users)	Only predefined users can see data. But every predefined user can validate transactions. e.g. a hybrid systems (Glaser 2016), or a private instance of a permissionless blockchain protocol such as Bitcoin or Ethereum	Only predefined users can see transaction data. Among those, only users with special rights can validate transactions. e.g. **Hyperledger Fabric**

Source: Friebe (2007)

This is not a closed cataloging of the blockchain. The importance comes when designing and building the ledger, whether to take into account if it is preferred more or less access and permission to processing data.

1.3. How does blockchain technology work?[8]

To fully understand how this technology operates, it is worth comparing it to the traditional way of registering transactions. In this way, for instance, it is common to go to the Land Registration to determine whether the seller of a house is the actual owner and legitimated to sell the property; similarly, when we purchase in our local grocery store using a debit or credit card, money is transferred from our bank account to the creditors. In both examples we are facing **centralised** and **black-boxed ledgers**. The first term means that there is a manager or middleman (public authority or bank) in charge of the ledger, mediates each transaction and whom all parties trust. On the other

[7] Friebe, T. (2017). *Is Blockchain Equal to Blockchain?*. [online] Medium. Available at: https://medium.com/blockchainspace/2-introduction-to-blockchain-technology-eed4f089ce5d [Accessed 16 Jun. 2018].

[8] European Parliamentary Research Service (2017). *How blockchain technology could change our lives*. [online] STOA, p.5. Available at: http://www.europarl.europa.eu/RegData/etudes/IDAN/2017/581948/EPRS_IDA(2017)581948_EN.pdf [Accessed 16 Feb. 2018].

hand, and as a consequence of the first, it is black-boxed or not fully accessible by the parties.

In this scenario, blockchain pretends to maintain the ledger of transactions, but with the innovation of completely erasing the centralised component. The immediate question that follows is how can a transaction be legitimate if there is no middleman that can validate and preserve it? The answer reveals the fundamental idea of blockchain: the database or ledger instead of being centralised, becomes **decentralised**. This way, each user of the system has a copy of the ledger and no longer needs a central authority. Thus, all users will have to verify the legitimacy of the transactions. Any user can request that a transaction is to be added into the blockchain, but before that happens each user must have had previously agreed upon[9], following the example, the seller is the owner and able to proceed with the sale.

The transactions agreed on will be recorded into a "block" and becomes the latest one in a chronological "chain" of blocks, and therefore the name of blockchain. These blocks are created by ***miners***, which are nodes or specialised computer hardware connected to the network in exchange for an economic revenue[10] or new currency (like bitcoins). Mining, therefore, is the process in which miners record and validate transactions that will be registered in the ledger. Anyone can be a *miner*, apart from any person connected to either party in the transaction to avoid possible conflicts of interest[11]. To be a miner implies facing competition between others to be the first to resolve complex mathematical problems and publish the next block. Once the block of transactions is updated into the blockchain, the other miners connected to the network must revise and validate all the transactions in the ledger[12].

---

[9] Crosby, M., Pattanayak, P., Verma, S. and Kalyanaraman, V. (2015). *Blockchain Technology Beyond Bitcoin*. [online] Berkeley: Sutarja Center, pp.5-8. Available at: http://scet.berkeley.edu/wp-content/uploads/BlockchainPaper.pdf [Accessed 13 Apr. 2018].

[10] Blockchain.info. (2018). *Miners Revenue*. [online] Available at: https://blockchain.info/charts/miners-revenue [Accessed 8 Apr. 2018].

[11] Of course, there's always a small chance this miner does know one of the persons involved in a recent transaction. Therefore, blocks are arranged in a chain: In roughly 10 minutes, when the next lottery winner is announced, this winner also confirms, as part of her announcement, that she agrees with all the transactions of the previous lottery winner

[12] EquiSoft (2017). *La cadena de bloques (blockchain) Una tecnología disruptiva con el poder de revolucionar el sector financiero*. [online] pp.3-4. Available at: https://www.equisoft.com/wp-content/uploads/2017/09/White-paper-Blockchain-ESP-1.pdf [Accessed 8 Mar. 2018].

It requires a unanimous approval so that the block that a miner created can finally be attached into the blockchain, but it is possible to accept the approval of a determined number of users according to the interests of the parties[13]. A block once added will permanently be there because it cannot be removed. This obeys the **immutability principle**, and it consists in not being able to modify or manipulate data once registered. Hypothetically, if a miner wants to change a transaction from history, he will have to remine every block till the current one, shown in every copy of the blockchain[14]. It requires not only the consensus of the participants but also a considerable computing power, so it is more theoretical than practical.

If we go back to the examples, destroying or corrupting the ledgers in a traditional transaction system by attacking the middleman (public authority or bank) is difficult, but not impossible in a world where hacking is growing in a vertiginous and dangerous manner. With the blockchain technology these threats became extremely difficult as every single user has their **own copy** of the ledger. Still, it is not immune to attacks or changes. The immutability depends on the permanency of the network, but the established consensus procedure could change by the community.

The blockchain process can be summed up as in Figure 1:

FIGURE 1: BLOCKAIN PROCESS IN A NUTSHELL[15]

Source: McLaughlin (2018)

[13] For instance, the Bitcoin blockchain requires a consensus of the majority (51%)
[14] Singh, A. (2018). *What makes a blockchain network immutable?* [online] Quora. Available at: https://www.quora.com/What-makes-a-blockchain-network-immutable [Accessed 5 Jul. 2018].
[15] McLaughlin, E. (2018). *How blockchain works: An infographic explainer*. [online] SearchCIO. Available at: https://searchcio.techtarget.com/feature/How-blockchain-works-An-infographic-explainer [Accessed 30 Apr. 2018].

1.4. Are blockchains programmable?

A ledger is created and designed to attend the needs of the parties. Thus, a blockchain is **programmable**, and all users will know what kind of data can be passed around and what must be rejected[16]. What is also programmable are the specific features of the type of blockchain (for example, choosing a private network in which only the main subsidiaries are permissioned to read and write into the blockchain), the extension of the concrete application of the blockchain (using a smart contract to sell only cars from Europe to Asia) and, of course, logical structures like "*if this, then that*" or "*else*" another outcome.

In its design there are some rules to be aware of in order to avoid conflicts among the users. There are **technical rules** (is there a consensus? Is the format of the information valid? Is there any missing required data?) but also rules inherent to the application of the blockchain, like **business rules** (do you have the right amount of cryptocurrencies? They are not being spent twice?) or **legal rules** (are you the legitimate owner to sell your property? What happens if the law changes?). Keep in mind that clarity is required because a blockchain can only do what it is programmed for and nothing else. That is why it is important to create a ledger that responds to the particular needs of the parties and its concrete application -smart contracts, in this Paper-.

1.5.What is inside a blockchain?

Reached to this point, we can understand what the blockchain technology is and how it functions in an abstract way. A blockchain is formed by a series of blocks, and each block contains data of transactions. This is how a block is structured[17] (Figure 2):

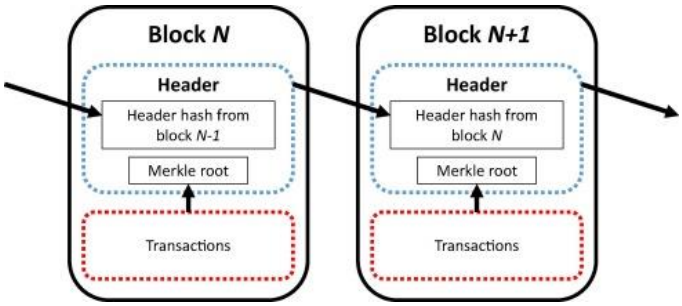
[16] Lewis, A. (n.d.). *A Gentle Introduction to Blockchain Technology*. Gentle Introduction Reference Papers. [online] BraveNewCoin, p.10. Available at: https://bravenewcoin.com/assets/Reference-Papers/A-Gentle-Introduction/A-Gentle-Introduction-To-Blockchain-Technology-WEB.pdf [Accessed 16 Feb. 2018].

[17] Dolader Retamal, C., Bel Roig, J. and Muñoz Tapia J. (n.d.). *La Blockchain: Fundamentos, aplicaciones y relación con otras tecnologías disruptivas*. [online] Cataluña: Universitat Politécnica de Catalunya, pp.33-35. Available at: http://www.mincotur.gob.es/Publicaciones/Publicacionesperiodicas/EconomiaIndustrial/RevistaEconomiaIndustrial/405/DOLADER,%20BEL%20Y%20MUÑOZ.pdf

- **Header**: includes metadata -data of other data- about: a) *Hash* of the previous block; b) *Timestamp* to identify the moment of creation; c) *Nonce* as the number miners are solving for; and d) *Root hash* to consult all the information of the block efficiently. This helps maintaining an order.
- **Content**: it depends on the application of the blockchain and it consists of a digital registration of statements or values. For example, if it is about bitcoin blockchain, the transaction of bitcoins would be the content; or, in smart contract, the selling of goods or services for a determined amount of currency (like ethers -Ethereum's currency-).

With time, the blockchain gets longer and the data it holds increases. Therefore, it is useful to count on a protocol that permits consult efficiently the information stored. For this purpose, a hash tree or Merkle tree is commonly used.

FIGURE 2: BASIC STRUCTURE OF A BLOCK[18]



Source: Sikorski et al. (2017)

1.6. How are users identified?

Blockchains are **potentially anonymous**[19] but are not necessarily this way. The good thing is that it provides trust by using encryption -binary values-. Who can participate in the network? Anyone containing the correct hardware[20] and software[21].

[18] Sikorski, J., Haughton, J. and Kraft, M. (2017). *Blockchain technology in the chemical industry: Machine-to-machine electricity market*. [online] ScienceDirect. Available at: https://www.sciencedirect.com/science/article/pii/S0306261917302672 [Accessed 8 Feb. 2018].
[19] Catchlove, P. (2017). *Smart Contracts: A New Era of Contract Use*. [online] Queensland University of Technology, pp.4-5. Available at: https://ssrn.com/abstract=3090226 [Accessed 31 May 2018].
[20] Generally, there are no special hardware requirements other than a simple CPU or any other electronic device capable of running the intended software (mobile phone, tablet, etc.). In other

Also, we must distinguish between public and private blockchains. On the first type, there is no such authority that controls the admission or checking identities; and when it comes to private blockchains, there is clearance to be granted and it is plausible that the authority who created the blockchain requires identification of the users.

The participants perform actions on the blockchain and these are stamped with digital fingerprints. The signature used in the platform is not electronical but digital. Electronic signatures refer to any sort of data that is electronically turned to sign a record or contract, like a handwritten signature or a digitalised image[22]. These are subject to be tampered and forged. As a solution, **digital signatures** are designed and are electronic records based on cryptography to authenticate a document from one digital space to another. The interesting factor is that it preserves the integrity of a document, states the parties who produced their signatures, but does not reveal the personal identity of the signatory[23]. In the blockchain, parties sign a hash which serves as a representation of the main document.

1.7. What do all blockchains have in common?

Public or private, permissioned or permissionless, created to support bitcoins, smart contracts or any possible application… we are talking about blockchains in any case. Yes, we can find differences, but the purpose is to understand the general bases of this technology. Thus, the main ideas that cannot be ignored are the following[24]:

---

cases, like mining on the Bitcoin blockchain, can require an upgrade using an ASIC hardware (Application Specific Integrated Circuits) to go beyond standard graphic cards.

[21] Such as Ethererum, SAP Leonardo, IBM Blockchain, Azure, Rubix, etc. It must be a computer program designed to be capable of interacting in the creation of a blockchain or its applications, working on a specific language (like C++, Solidity and Javascript).

[22] Latimer, P. (2011). *Signatures, Squiggles and Electronic Signatures*. [online] Swinburne University of Technology, pp.6-8. Available at: https://ssrn.com/abstract=1601169 [Accessed 29 May 2018].

[23] Thompson, S. (2017). *The preservation of digital signatures on the blockchain*. [online] University of British Columbia, pp.2-4. Available at: http://ojs.library.ubc.ca/index.php/seealso/article/view/188841/186525 [Accessed 31 Jun. 2018].

[24] Deloitte (2018). *What is Blockchain?* [online] p.7. Available at: https://www2.deloitte.com/content/dam/Deloitte/uk/Documents/Innovation/deloitte-uk-what-is-blockchain-2016.pdf [Accessed 4 Apr. 2018].

1. A blockchain is a **digitally distributed ledger** among various computers in practically real time. It is distributed to all users and participants, keeping each a personal copy of the whole ledger. By this way, the blockchain is decentralised and needn't of the trust of a middleman.
2. The network is formed by multiple participants that **reach consensus**. The trust in this system relies on that the relevant participants verify the legitimacy of each new block before it is added into the blockchain.
3. A blockchain is **based on cryptography and digital fingerprints**. This permits to identify the participants by their actions in the network.
4. The ledger is also **time-stamped**: this way transactions are easy to be tracked down and validated.
5. A blockchain follows the **immutability principle** of its data. The information recorded in the ledger cannot be changed.
6. A blockchain is **programmable**: according to the needs of the parties determining the information to use, type of blockchain, its specific application -smart contracts- and instructions that follow logical structures like "*if this, then that*" or "*else*" another outcome, as actions to be executed when certain conditions are met.

2. FUNDAMENTALS OF SMART CONTRACTS

2.1. What are smart contracts?

The concept of *smart contract* was introduced by Nick Szabo in 1996, conceived as "*a set of promises, specified in digital form, including protocols within which the parties perform on these promises*"[25]. It was not till 2009, when the technology was more developed and the bitcoin blockchain emerged, smart contracts started to develop, and especially from 2015 with the creation of Ethereum.

It is important to state that smart contracts are not just automated contracts, that already existed for a long time. Szabos' definition does not capture the difference

[25] Szabo, N. (1996). *Nick Szabo -- Smart Contracts: Building Blocks for Digital Markets*. [online] Fon.hum.uva.nl. Available at: http://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/smart_contracts_2.html [Accessed 13 Apr. 2018].

between merely automated contracts, like vending machines, which are programmed with certain rules that can be included in a contract and respond to those rules (for example, if you insert 1€, a can of Coke will drop). The difference is that smart contracts are addressed in a decentralized network that automates the performance, which can be the reason for it being called "smart".

The definition of a smart contract has not reached consensus and there are multiple approaches[26], which is understandable due to the nature of this new phenomenon and the complex technology it involves. In this Article, we will work on a concept that most authors contemplate: "*digital contracts allowing terms contingent on decentralized consensus and are self-enforcing and tamper-proof through automated execution*"[27]. In a simple way, a smart contract is an agreement between two or more parties, "*encoded in such a way that the correct execution is guaranteed by the blockchain*"[28]. Note how these definitions include the use of a decentralized ledger, and not simply a digital contract between parties and written in a computer language. This way of viewing is consistent with Szabos' approach but goes beyond.

However, a smart contract does not need a blockchain to function[29], but it is relevant due to the security features it presents -immutability and digitally distributed among users-. That is why it is standard to use blockchain, in particular Ethereum, as a decentralized execution platform that stores smart contracts[30].

Other authors point out that a smart contract is not a contract, nor intelligent, and proposes a different term: Programmatic Executable Transactions (PETs). We can say that there are values agreed on by the parties to be executed, as a sequence of code and data. It does not seem wrong to assume it is not a contract because it is a software, but the purpose of the term "contract" based on the intent of the parties to program their

[26] Stark, J. (2016). Making Sense of Blockchain Smart Contracts. [Blog] *Coindesk*. Available at: https://www.coindesk.com/making-sense-smart-contracts/ [Accessed 7 May 2018].
[27] Cong, L. and He, Z. (2018). *Blockchain Disruption and Smart Contracts*. [online] pp.11-12. Available at: https://ssrn.com/abstract=2985764 [Accessed 4 Jun. 2018].
[28] Wattenhofer, R. (2016). *The science of the blockchain*. 1st ed. Inverted Forest, p.87.
[29] Nothing stops creating a smart contract embedded into a traditional database, but then the parties would rely on a trusted centralised party and the ledger will not be as immutable as using a blockchain. By this way, it loses its sense of "smart".
[30] Bashir, I. (n.d.). *Mastering blockchain*. Packt, p.103.

terms and values and, more importantly, to create smart contracts as an alternative to traditional contracts.

That being said, the term smart contract is not the same used in computer or legal language. According to the first one, it is not *per se* a "contract" yet can be an agreement in a computer programming perspective. For it to be considered valid it must comply with the legal requirements -consent, object, cause and form-.

Making it simple, **a *smart contract* is a software** -call it contract or not- **that permits the automated execution of an agreement contained directly in the smart contract itself or acting as enforcement of a conventional contract, and recorded on the blockchain.**

2.2. What are its characteristics?

Out of the definition proposed, to make it more understandable, it is worth highlighting its core features, which are[31]:

1. **Electronic nature.** Typically, a contract is created written or orally. Also, with the development of e-commerce, it is frequent to distinguish electronic contracts, though it can require paperwork like receipts o invoices as proof of this contract. Unlike these, a smart contract cannot exist but in electronic form**.** Moreover, it is linked to electronic data -to be self-enforceable- and rests on digital signatures based on cryptography. Nevertheless, celebration of a smart contract and its performance can be off-chain and, therefore, not electronic.
2. **Software-implemented.** The idea is that "*code is law*". The contractual terms are established in a software with computer codes. Therefore, smart contracts not only regulate the relations of the parties but also is a computer program according to Intellectual Property law. It will be created by the demand of the parties and the subsequent subscribers.
3. **Increased certainty**. If a conventional contract -oral or written- is interpreted by humans, a smart contract is formed by computer codes that are interpreted by

[31] Savelyev, A. (2016). *Contract Law 2.0: «Smart» Contracts As the Beginning of the End of Classic Contract Law*. [online] National Research University Higher School of Economics, pp.11-16. Available at: https://ssrn.com/abstract=2885241 [Accessed 10 Mar. 2018].

computers itself. Programming these codes has the advantage of being precise so that all parties can predict the outcome of the contract. There still can be ambiguities, especially regarding what has not been agreed on, but it provides more certainty than common contracts.

4. **Conditional nature**. Computer codes follow the logic of "*if this, then that*". Parties will establish their terms using a conditional statement that will enforce the contract.
5. **Self-enforceable.** Means that once the smart contract is agreed on and running, the execution of its codes is automatic and will not require a specific approval. The parties (or even third parties), thus, do not have any power to stop this process, even if they change their mind and fall into programming errors. For instance, once agreed on a payment transfer every first Sunday of the month for the next 5 years of 10.000€, signifies that for the next 5 years the transfer will be affected on that specific day and for that amount. This feature also increases the certainty of the smart contracts.
6. **Self-sufficient.** The existence of a smart contract functions by the computers given rules and, in principle, even if it seems immoral or against the law, which gives rise for controversy and demands certain action to avoid an illegal execution.

As it can be seen, these elements are all connected and dependant on each other to conform a smart contract

2.3. What are the benefits and risks?

The main benefits and risks[32][33] can be inferred from the definition of smart contracts, and not only for consumers and businesses but also for public authorities. In comparison to conventional contracts, smart contracts are beneficial by:

[32] Chandler, R. (2016). *Smart contracts*. Wroclaw: Amazon Fulfilment.
[33] Ream, J., Chu, Y., Schatsky, D. (2016). *Upgrading blockchains: Smart contract use cases in industry*. [Blog] Deloitte Insights. Available at: https://www2.deloitte.com/insights/us/en/focus/signals-for-strategists/using-blockchain-for-smart-contracts.html [Accessed 15 Mar. 2018].

1. **Certainty**. Parties celebrate traditional contracts to provide certainty, but there is room for disagreement and breach. Smart contracts prove higher levels of certainty in two ways: a) it is **accurate** thanks to its less ambiguous logical formulation "*if this, then that*", reducing human error in writing and reading; b) **verifiable** because it will be encrypted on a ledger, having everyone the same copy and undeniable of its existence and the terms agreed on.
2. **Autonomous**. When the middleman is eliminated, the smart contract takes the control of the terms and the execution is automatic by the network, avoiding manipulations.
3. **Speed**. Instead of manually filling out the contract and additional paperwork, the use of a software code automates these tasks. Also, updates are inserted in real time.
4. **Lower costs**. Savings come from reducing the needed time to fill a traditional contract, money to be paid to employees to complete these tasks, avoiding future costs by reducing error and, especially, the intermediary to validate and execute the contract.
5. **Security**. Smart contracts and its data in the decentralized registry will be safe using cryptography and encryption. They cannot be lost -every party has a copy- and extremely difficult to be hacked. If it is the case and a malevolent enters the blockchain, by using arbitrary addresses, he will not be able to access personal information.
6. **New businesses or operational models**. The characteristics of smart contracts and cost reduction prove to be a way of enforcing a contract and, therefore, allow new possibilities. For instance, electric cars can recharge by induction while stationary in some roads or at traffic lights using smart contracts[34]. This example fits in what is known and the **Internet of Things (IoT)**, a system that interconnects computer devices with Internet access (like cars, kitchens, heart monitors, etc.) and transfers data over a network without needing a direct human intervention. A smart contract can execute its terms and interact with digitally connected devices.

As always, there is a downside to consider. Main disadvantages are:

[34] Such as RWE, that pretends to use a smart contract to communicate with electric car stations so that the users can rent the station and charge their car.

1. **Unconvincing**: the early stages of development of the blockchain and smart contracts pushes back consumer, companies and public authorities. The risks that it involves, the complexity of these technologies and hardly any previous references make users suspicious. People are used to writing documents that regulate parties' rights and obligations and sign it.
2. **Errors**. Yet very certain, if the code is not written precisely to the intention of the parties or simply correctly in programming language, the system will not execute as intended in the first place.
3. **Inflexibility**. The whole idea is to agree on terms to be are auto-executable. But, what happens if there is a change of mind? Or the terms are badly expressed? Parties must anticipate future scenarios that can require changes.
4. **Third parties do not disappear**. They will play a new role, for example, experienced lawyers in IT can advise their clients in the making of new contracts.
5. **Contractual secrecy**. Traditional contracts keep the information within the parties; but as smart contracts are executed on a distributed ledger, the users could be aware: the information processed by the smart contract must be decrypted, and the codes are executed in all nodes of the network[35]. This problem minimizes in permissioned blockchains, to grant access only to relevant users.
6. **Latency**. It takes time to time for each block to be verified and added into the blockchain, compromising updates.
7. **Uncertainty in regulation**. How smart contracts will respond to the law is not fully clear. That is why its recognition by legal authorities can be critical for the development of some applications to avoid legal consequences or assuming business loss.

[35] Staples, M., Chen, S., Falamaki, S., Ponomarev, A., Rimba, P., Tran, A., Weber, I., Xu, X. and Zhu, J. (2017). *Risks and Opportunities for Systems Using Blockchain and Smart Contracts*. [online] Data61 (CSIRO), p.42. Available at: https://publications.csiro.au/rpr/download?pid=csiro:EP175103&dsid=DS2 [Accessed 31 May 2018].

2.4. What can it be used for?

Smart contracts are becoming the center of attention due to all the possibilities it can provide. They can practically be used in any scenario in which it is intended to transfer or store secure and unalterable data without intermediaries. The following Table 2 schematizes these uses:

TABLE 2: POSSIBLE APPLICATIONS FOR SMART CONTRACTS[36]

Use case		What the smart contract can do
Financial services	Trade clearing and settlement	Manages approval workflows between counterparties, calculates trade settlement amounts, and transfers funds automatically
	Coupon payments	Automatically calculates and pays periodic coupon payments and returns principal upon bond expiration
	Insurance claim processing	Performs error checking, routing, and approval workflows, and calculates payout based on the type of claim and underlying policy
	Micro-insurance	Calculates and transfers micropayments based on usage data from an Internet of Things-enabled device (example: pay-as-you-go automotive insurance)
Life sciences and health care	Electronic medical records	Provides transfer and/or access to medical health records upon multi-signature approvals between patients and providers
	Population health data access	Grants health researchers access to certain personal health information; micropayments are automatically transferred to the patient for participation
	Personal health tracking	Tracks patients' health-related actions through IoT devices and automatically generates rewards based on specific milestones
Technology, media, and telecom	Royalty distribution	Calculates and distributes royalty payments to artists and other associated parties according to the contract
Energy and resources	Autonomous electric vehicle charging stations	Processes a deposit, enables the charging station, and returns remaining funds when complete
Public sector	Record-keeping	Updates private company share registries and capitalization table records, and distributes shareholder communications
Cross-industry	Supply chain and trade finance documentation	Transfers payments upon multi-signature approval for letters of credit and issues port payments upon custody change for bills of lading
	Product provenance and history	Facilitates chain-of custody process for products in the supply chain where the party in custody is able to log evidence about the product
	Peer-to-peer transacting	Matches parties and transfers payments automatically for various peer-to-peer applications: lending, insurance, energy credits, etc.
	Voting	Validates voter criteria, logs vote to the blockchain, and initiates specific actions as a result of the majority vote

Source: Deloitte Insights (2016)

[36] Ream, J., Chu, Y., Schatsky, D. (2016). Upgrading blockchains: Smart contract use cases in industry. [Blog] *Deloitte Insights*. Available at: https://www2.deloitte.com/insights/us/en/focus/signals-for-strategists/using-blockchain-for-smart-contracts.html [Accessed 15 Mar. 2018].

III. SMART CONTRACTS AND CONTRACT ELEMENTS

This section is dedicated to answer if a smart contract is really a contract from a legal perspective. For that, it is necessary to see what the law – Spanish Law- has to say. It is clear that there is no reference to smart contracts, so being a contract or not depends on its response to essential elements and form.

1. ESSENTIAL ELEMENTS

For what has been said, a smart contract is basically a computer program that produces automatically determined consequences once a condition is triggered. Legally, for it to be a contract, it must follow the requirements established in the applicable law. According to the Spanish Civil Code -from now CC-, all contracts include: 1) consent, 2) object and 3) cause (art. 1261 CC).

1.1. Consent

Out of the three, clearly, consent is the most difficult one to approach. The problem underlies the nature of this technology[37]:

a) The self-enforcement and self-sufficient features of the smart contract can produce **consequences that a user does not really understand or want**, somehow forced. It is possible to face **vices of consent** and, therefore, invalidate the consent. For that, *"it must refer to the substance of the thing that is the object of the contract, or on those conditions thereof that mainly gave reason to celebrate it*" (art. 1266).

b) Blockchain technology intends to keep secrecy of the identities of the users, so that if a hacker enters in the system, he will not be able to extract any personal data -only arbitrary addresses-, which can also **enable those whose are legally not permitted to consent or incur in a legal prohibition** (arts. 1263 y 1264 CC).

How is consent expressed in a smart contract? According to the Civil Code, **consent is manifested by the concurrence of the offer and acceptance** (art. 1262.I CC). Therefore, when two parties agree expressly determined terms in a smart contract to

[37] Navas Navarro, S., Robert, S., Górriz, C., Castells i Marquès, M., Camacho, S. and Mateo, I. (2017). *Inteligencia artificial*. Valencia: Tirant lo blanch, pp.192-193.

regulate and execute it, under the tender of "*offer and acceptance*", then it will be a contract. Tacitly, this is, through conclusive and unequivocal acts, would also be valid. In any case, the consent must clearly determine: a) What is the performance parties must do; b) How, when and why of its realization; c) And what are the consequences of not doing so[38].

1.2.Object

The object of the obligatory relationship is the agreed benefit, which must be **possible, lawful and determined** (arts. 1271 to 1273 CC), and it consists of an obligation to give, do or not do something (art. 1088 CC). It applies naturally to a smart contract in the same way. The particularity, in this case, is that it is a new technology and way of doing things that enable new possibilities -see I. 2.4. *What can it be used for?*- that are not fully known yet. What is clear is that it cannot serve to violate the laws or trade things out of commerce.

1.3. Cause

It is an essential element and, like the object, does not give rise to further problems respect a conventional contract. It is necessary an **existing cause and to be lawful** (art. 1275 CC). The celebration of a smart contract cannot be an incentive to go against the law, like simulating inexistent operations or escaping tax payment.

2. FORM

The principle of private autonomy in Spanish law (art. 1255 CC) is the power conceded to an individual to govern its own legal sphere. It parts as a superior value of the legal system (art. 1.1 Spanish Constitution, -from now on CE-), as an expression of liberty and, in particular, as a manifestation for the sake of the free will to develop our personality (art. 10.1 CE).

A smart contract can be a contract as far it can fit in its definition and, according to the Civil Code, "*The contracting parties may establish the covenants, clauses and conditions that they deem convenient, provided they are not contrary to the laws,*

[38] Tur Faúndez, C. (2018). *Smart contracts*. 1st ed. Madrid: Reus, pp.83-84.

morals or public order" (art. 1255 CC). That means that Spanish Law establishes **liberty of form** as a general rule to contract, as long as the law is respected, but only the imperative norms and not the dispositive. It must be said that, in a law like ours where private autonomy is a general principle, dispositive norms are the majority so that people can regulate their private relations in an equal position. So, in this way, if the parties have agreed on using a smart contract on an existing contract, they can reciprocally compel each other to follow these terms (art. 1279 CC).

Nevertheless, the form acts as a **requisite of efficacy**. Smart contracts use specific language that does not only prove its existence -like in other type of contracts-, but also determines its efficacy to the point of making it lose its effects (such as not being automated when the codes are not correctly formulated) or making the performance impossible (by not concreting actions or benefits of the parties, the codes cannot produce a not programmed outcome). This means that we would not be in front of a smart contract and its natural effects, but it does not deny the validity of a previously existing contract[39].

On occasions, the balance turns out decompensated for one of the parties, which hold a preeminent position over the other. In these cases, imperative norms take place to protect the weak subject of the relation. There are two laws of great importance in Contract Law that may generate uncertainty when applied to smart contracts:

a) Law 7/1998, of 13 of April, of General Conditions of Contract.

These general conditions are clauses predefined to be incorporated in the contract and imposed from one party to the other, created to be used in multiple contracts. To be valid, in a contract -and in a smart contract- it is necessary the fulfillment of some requirements. Here we highlight the following:

The conditions will be part of the contract once accepted by the adherent and signed by all parties. Previously, the **adherent must have been informed about its existence**

[39] Feliu Rey, J. (2018). Smart Contract: Concepto, ecosistema y principales cuestiones del Derecho privado. In: *La Ley mercantil: Contratación mercantil*, 47, ed. Wolters Kluwer, pp.7-10.

and facilitated a copy (art. 5.1). It is key that both parties have clear, precise, comprehensible and enough information so that there are aware of the automated execution process and its consequences regarding their agreement and patrimonial sphere. Moreover, the **clauses will adjust to criteria of transparency, clarity, concretion and simplicity** (art. 5.5). If not, the aggrieved party can exercise the nullity action (art. 1301 CC).

It can be imagined creating multiple smart contracts to regulate relations with different subjects using these general conditions of contract with computer codes. We do consider that there would not be any problems applying this Law to smart contracts as long as its norms are respected.

b) General Law for the Defence of Consumers and Users, approved by the Royal Legislative Decree 1/2007, of 16 of November.

There is also a weak party, but in this case, it is a consumer or a user, while the predominant subject is a businessman (art. 2).

This Law includes various precepts to protect consumers and users and it is to highlight two general ones: "*The previous renunciation of the rights that this norm recognizes is null*" (art. 10); and "*All non-negotiated stipulations will be considered abusive clauses individually and all those practices not expressly consent, against the requirements of good faith, to the detriment of the consumer or user, an important imbalance of the rights and obligations of the parties arising from the contract*" (art. 82).

Other norms to take into account are that the businessman must facilitate in a clear and comprehensible manner before it attaches the consumer or user, unless it is understood by the context with relevant, truthful and sufficient information, including inalienable rights (art. 60); the necessity of unequivocal will of contracting and how to end the relation (art. 62); and the delivery of a receipt and invoice (art. 63), among others. In any case, these imperative norms demand a case-by-case study to verify if there is an abuse over the weak party. It is natural to also demand these requisites in smart contracts.

IV. PHASES OF SMART CONTRACTS

Determined that a smart contract can comply with the requirements that the Spanish law stipulates, now is the moment to analyse the main problems that this contract can present in its different phases. From now on smart contracts will be studied in three stages: 1) Formation and perfection; 2) Performance and modification; and 3) Breach.

1. FORMATION AND PERFECTION

1.1. When is a smart contract binding on the parties?

The **formation** of a contract comes before its perfection and, therefore, it is aimed for the parties to reach an agreement and be attached to their obligations. The conversations and deals in this part do not oblige the parties, but an arbitrary rupture can cause extracontractual responsibility. In this phase, the preliminary deals are placed to negotiate if the smart contract is the contract itself or derivative of a previous contract, establish their intentions, order studies to experts, etc. This is important in smart contracts because it is a new technology and parties must be precise in their intentions and properly reflect it in the software so that certainty can display.

A contract is perfected in the exact moment it exists and links the parties to comply with the obligations agreed on unless there is an accidental element (term or condition) that delays or suspends its effects. If before was said that the consent is expressed by the concurrence of the offer and acceptance, here we determine that the **perfection of a smart contract comes with the mere consent** (art. 1262.I CC), as the general rule in Spanish law. Therefore, this is when a smart contract becomes binding on the parties. Knowing that agreeing on using a smart contract is most probable that the location of the offer and the acceptance are not situated in the same place, there is consent "*since the offeror knows the acceptance or since, having been sent by the acceptor, cannot ignore it without missing the good faith*" (art. 1262.II CC).

This stage is not very different between conventional contracts to smart contracts. In both cases, the parties will agree on the terms that will regulate their rights and obligations. The difference lies in the fact that a smart contract can act as it is the

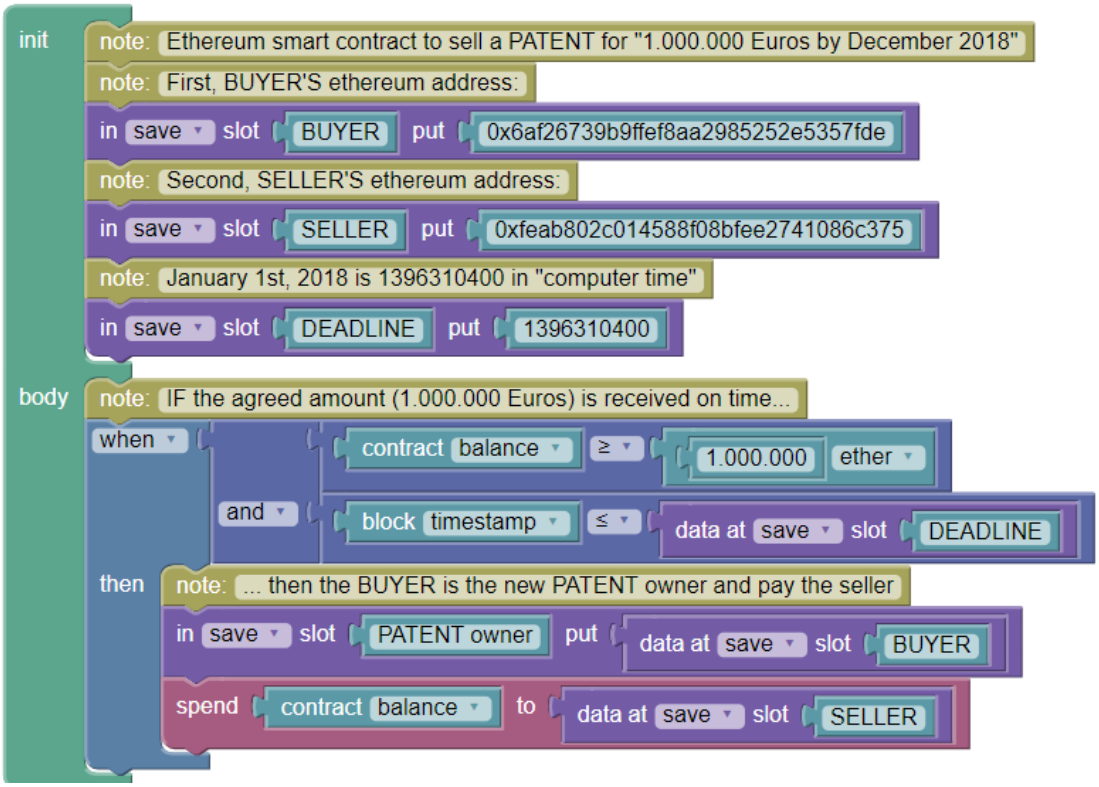
contract itself or it could be derivative of a previous written contract, but until the program initiates there is no smart contract. In addition, some authors consider a possible mixed version or hybrid, where conventional and smart contracts will contain different terms, yet connected.

Whether one option or another is chosen, a smart contract can be incorporated in the ledger as an offer (declaration of will done to another person proposing celebration of a contract with intent to be obliged by it, with all the necessary essential elements), pending of acceptance (declaration of will by the receptor of the offer to manifest his complete agreement) for the formation of the contract, yet not perfected.

To illustrate in a basic way what a smart contract would look like (Figure 3), let's say that Mr. X has offered, and Mrs. Y has accepted to acquire a license to exploit a patent regarding human genes. Now, two parts can be differentiated:

1) **Initialization**. This space is dedicated to formalizing the Ethereum address of both, seller and buyer, and the deadline (1st January) to receive the payment (1.000.000 €). Incorporated the consent, the smart contract is then perfected and binding on the parties.
2) **Body**. This section incorporates the "*if this, then that*" logic: if the buyer receives the money on time (1.000.000 € or higher) before the deadline (1st January), then the buyer becomes the new owner of the patent and pays the seller. Participants must pay for fees in form of Ether, as a currency value, to record a transaction in the Ethereum blockchain.

FIGURE 3: BASIC SMART CONTRACT STRUCTURE



Source: Own creation using www.etherscripter.com

1.2.Can there be a change of mind?

Another situation to question is what happens if one party wants to change their mind. When parties agree on certain terms they must comply with them and, with smart contracts, there is no option because the execution becomes automatic. Nevertheless, there is another possibility: that the parties realise that in the future their mind can be changed under certain conditions and, therefore, configure it in a way that allows it.

Say the manager of a real estate company hires professionals and celebrates a smart contract that stipulates a reward to those who manage to sell at least four houses per year. He decides that the income will be increased by 20% of the earned profit for these employees. This promise once offered and accepted force the reward if the condition applies. In this way, whatever happens, the company will reward whoever sells more than four houses. Either it is explicitly written that there is no change of mind or simply not contemplated this possibility in the computer program, in any case, it is **not changeable**.

Picture, in this case, **before perfecting the smart contract**, the manager of the company realises that if a negative scenario takes place they would be in deep debts and might fall into bankruptcy. Now, the smart contract would incorporate the right to change mind. How would it be done? The problem is solved by allowing both parties to **codify in the computer program a certain finality** so that, though not knowing for sure the future outcome, parties can "*organize their behaviours around a mechanical certainty or lack thereof*"[40]. In this example, the agreement would have an increment 20% of the earned profit for employees who sell at least four houses per year, and without compromising the companies' financial situation.

1.3. Is there room for misunderstandings and mistakes?

Smart contracts promise to be more efficient than traditional contracts lowering costs, being more secure, faster and, of course, in certainty. Computer language is logical and uses fewer terms -and consequently reduced meanings- than what a human would recognize. This is how it has the potential to minimize conflicts.

Ambiguity does not disappear from computer programming, but it is for sure much less than the existing in the real world. If the language used in conventional contracts is infinite, the codes that configure a smart contract must be predefined. Whatever is not determined is a "nonsense" for the program because it cannot understand human language, but only the codes that it is programmed to interpret and execute. Also, it will not apply to any computer language but only the one the software is prepared to deal with -for example, *Solidity* is the language implemented in Ethereum-.

A smart contract can be executed in a manner that the parties did not exactly intend. When established the codes to regulates the obligations, **nobody can claim "I do not understand" what is being recorded because it will be executed automatically**. This reduces ambiguity and, in consequence, provides certainty by predicting what will happen in the future according to terms agreed on. It is necessary to understand this feature or limitation of a smart contract in order to its formation and perfection according to the true will of the parties.

[40] Raskin, M. (2017). *The Law and Legality of Smart Contracts*. 1 Georgetown Law Technology Review 304. [online] p.325. Available at: https://ssrn.com/abstract=2959166 [Accessed 15 Apr. 2018].

2. PERFORMANCE AND MODIFICATION

From formation to execution, a correct approach to a smart contract would include terms and conditions by the parties that, once the determined events are triggered, the contract is executed and settles the object agreed on in the blockchain (like transferring or earning cryptocurrencies after selling a car) or in the physical world (getting hold of a real car). Nevertheless, the performance and modification can suggest certain doubts that must be answered.

2.1.What happens if the performance is defective?

In Spanish Law, a contract is valid, respecting its essential elements – consent, object and cause-, despite the execution not being flawless and, therefore, still obliges the parties. Moving this situation to smart contracts, it is imaginable the production of an outcome that parties were not expecting due to not translating their will in code exactly how the result should be. Like it was just seen, it is very important that they establish the terms in a precise way to reduce ambiguity. At this point there are two options: **1)** Not only being concrete in the codes but also **establishing a margin for discretion**; **2)** Or simply **using a conventional contract**, admitting that a smart contract is not recommended for situations in which discretion is an important factor for the parties. The following shows an example of an obligation that requires discretion:

> *Mr. X orders Mrs. Y the creation of a sculpture of himself. The debtor assumes an obligation of means -not of result-, in which she complies acting according to the diligence characteristic of her art. The complete satisfaction of the buyer is not guaranteed, and a computer program cannot recognize a not perfectly defined outcome.*

2.2. Can performance be withheld?[41]

The reality a smart contract tries to regulate can be much more complex than a simple "*if this, then that*" structure -even though it is core- and just automate a

[41] Tjong Tjin Tai, E. (2017). *Formalizing Contract Law for Smart Contracts*. Tilburg Private Law Working Paper Series No. 6/2017. [online] Tilburg Law School, pp.6-8. Available at: https://ssrn.com/abstract=3038800 [Accessed 3 Jul. 2018].

determined consequence. If Contract Law recognises the right to withhold performance when the other party has not completed its obligations as stipulated, smart contracts are not intended to do so because of the self-enforceability feature. Nevertheless, smart contracts should be prepared to distinguish between events that comply automatically (when a certain event triggers the codes of a smart contract and by itself satisfies a predefined outcome) to those that need an off-chain verification (when a smart contract itself is not enough to produce a predefined outcome and needs external input of data to proceed). To illustrate this point, it is worth an example:

The local Car Company delivers a car to Mr. X agreeing on a payment due every month.

A) Scenario 1: When the condition is met automatically. The creditor (Car Company) that must perform a reciprocal obligation (deliver the vehicle for its price) at the same time or after the debtor (Mr. X), has the right to withhold his performance pending on the compliance of the debtor. So, when the debtor does not pay on time he incurs in **breach** -non-performance- and the procedure of the smart contract ends.

Say that the car is specially programmed to receive instructions linked to a smart contract and if the debtor does not pay, the car will not start (Algorithm 1).

Algorithm 1 Payment enforcement

```
1: procedure ENFORCE(p, term, car)
2:     if p < term then                ▷ Term fully paid?
3:         InterruptStarter(car)
4:     end if
5: end procedure
```

Source: Tjong Tjin Tai (2017)

B) Scenario 2: When the condition requires an offline verification. This possibility must be expressly permitted by the smart contract. Following the example, suppose:

Mr. X can withhold payment of a term if the car is accidented before the sale or after by being a damaged product. The reason is to protect the buyer.

The buyer will withhold the payment in **anticipation of breach**[42] **–** having reasons to believe that the seller has not delivered the car in immaculate condition and decides to not continue paying monthly- and must notify the seller of the situation of the vehicle (Algorithm 2). If the smart contract is not prepared to check by itself if a product is defective, then the parties will need to find a way to solvent this problem by themselves assuming the word of one of them or require the intervention of a third party – See *IV. 3. Breach and remedies*-.

Algorithm 2 Payment enforcement extended

```
procedure ENFORCE(p, term, car)
    if p < term then                          ▷ Term fully paid?
        if WithholdingAllowed(car) = FALSE then
            InterruptStarter(car)
        else
            NotifySeller(car, PaymentWithheld)  ▷ Notification is required
        end if
    end if
end procedure
```

Source: Tjong Tjin Tai (2017)

The number of reciprocal obligations and their complexity will determine the structure of the smart contract. To keep it simple, we assume the reciprocity consists in delivering a car by the Car Company and paying for it by Mr. X. The general scheme of withholding performance would be (Algorithm 3):

Algorithm 3 Withholding performance

```
procedure WITHHOLDINGALLOWED(car)
    if NonPerformance(car) = TRUE then
        return TRUE
    else
        return FALSE
    end if
end procedure
```

Source: Tjong Tjin Tai (2017)

Basically, this algorithm establishes that if one party does not perform as expected, the other may withhold payment. Truly, it is not that simple because:

[42] Even though the buyer will not breach because the vehicle is defective, the smart contract interprets not paying the creditor as a breach, unless the damage is notified.

- **A general rule for performance is complicated** when reciprocal obligations must be defined. To simplify the programming, it is recommended to only allow specific cases.
- **The rule will have to be checked by a third party** to verify the real-world events. If Mrs. Y claims the car being a defective product, it needs to be revised by a trusted party.
- Another possibility is to simplify the contract by **erasing this rule**, though it means removing the buyers' protection.
- **The law can entail additional requirements**, like to notify the seller of the damage and sending guarantee documents.

With all that said, one of the disadvantages of smart contracts is that programming it can be tough for certainty in execution. The basic logic code of "*if this, then that*" can be subsumed in another "*if this, then that*"; and more codes can be added like "*return*" the money. An extended example could be (Algorithm 4)

Algorithm 4 Withholding performance full

```
procedure WITHHOLDINGALLOWED(car)                    ▷
First part is (1), after breach
    if BuyerComplaint(car, breach) = TRUE then
        if NonPerformance(car) = TRUE then
4:          return ReasonableAmount(term, Facts)
        else
            NotifyBuyer(CorrectPerformance);
            return 0
8:      end if
    else                    ▷ Now anticipatory breach (2)
        if BuyerAnticipatoryComplaint(car, Facts)
= TRUE then
            if Reasonable(car, Facts)=TRUE then
12:             return
ReasonableAmount(term, Facts)
            else
                NotifyBuyer(UnreasonableWithholding);
                return 0
16:         end if
        else
            NotifyBuyer(NoNotification)
            return 0
20:     end if
        return 0
    end if
end procedure
```

Source: Tjong Tjin Tai (2017)

2.3.Can a smart contract be modified?

This is a controversial question that threatens the revolution a smart contract pretends over the traditional way of doing things. What has been said till now is that once determined the terms on the computer program, there is no going back because the key feature here is the auto-enforceability and immutability.

However, it seems sharp to be generalized for all types of contracts and especially **when an outcome against the law is forced**. Think about a smart contract in which the debtor must conserve some goods to be seized by the creditor after 60 days. Sometime later, the law changes and establishes retroactively a minimum of 120 days. The contract was correctly formed but turned out to be against the law due to its change. Would the smart contract continue to execute automatically as originally agreed on and be, therefore, contrary to law?

There are different ways of facing this problem, from a government solution to a private one. Some methods to be considered are:

1) One option would be *ex-ante*, for the creation by the public authorities of a new infrastructure known as an **Application Programming Interface (API)** and public database that collect important legal provisions. By this way, when parties create their smart contract, it would be able to recognize the legal updates in the database and thereupon update the smart contract terms.
2) Another chance, *ex-post*, would not employ the need for government to create an API because the **parties rely on themselves** manually to police the smart contract. The disadvantage is that there could be a party that tries to impose changes to their interest. To minimize this worry, it is worth to determine what terms can be modified (for instance, the payment) and those that cannot be touched under no circumstance (like the period of time the debtor has before enters into the breach).

However, what seems more common is the **amendment of a smart contract by the will of the parties**. Will this be possible? At the beginning of this Paper the immutable principal was remarked -see II. 1.3 *How does blockchain technology work?*- and we

know that contracts attached to a blockchain stay permanently there. As a solution[43] we can create a:

1) **New smart contract but same address**. Here we apply the same principle, but parties can update the present smart contract (SC0) creating a new one (SC1) with the same address as the first (SC0) in the blockchain. By this way, SC0 will execute SC1[44], and this one will execute its own codes.
2) **New smart contract and new address**. This would consist in extracting the information of SC0[45] and creating SC1 as a new one with the same data but altering the code that is meant to be modified, and in a new address. This is a more radical option being a totally new smart contract.

2.4. What are oracles?

Blockchain and smart contracts are decentralized and needn't of a third party, making this phenomenon so special. And it is true, but facing complex circumstances obliges the involvement of a third party in some situations, like an oracle.

A smart contract is prepared to verify conditions that exist in the blockchain, but its self-sufficiency is not capable of jumping into to the real world and checking what information is correct or not. Here comes the **oracle, an external platform**[46] **to the blockchain that updates the smart contract with information for its intended execution** (following the previous example, if a product is defective, and other sort of data: foreign exchange, share market values, if a team A or B won, etc.). This information can come from[47]:

a) **Unquestionable facts**: when there is no room for interpretation and is always the same no matter the source and, therefore, do not cause any objection to the

[43] Grincalaitis, M. (2018). *Can a Smart Contract be Upgraded/Modified?* [online] Medium. Available at: https://medium.com/@merunasgrincalaitis/can-a-smart-contract-be-upgraded-modified-1393e9b507a [Accessed 4 Jul. 2018].
[44] In Solidity the "*delegatecall*" opcode will be used.
[45] In Solidity SC0 can be eliminated using the "*selfdestruct ()*" command.
[46] This external agency can be a company, institution or a database to resolve details which cannot be known at the time of perfection of the smart contract cannot.
[47]Feliu Rey, J. (2018). Smart Contract: Concepto, ecosistema y principales cuestiones del Derecho privado. In: *La Ley mercantil: Contratación mercantil*, 47th ed. Wolters Kluwer, pp.15-16.

parties. For instance, a smart contract that will execute the payment of dividends if the shares of our company increase by 20%, can be linked to a database (New York Times, BBC, etc.) that will all refer to the share market values.

b) **Interpretable facts:** when a smart contract needs information that requires a value of judgement. It is important to allow only impartial agencies to intervene to avoid going against the law and keeping in mind the immutable principle. For example, different experts can disagree with considering if the delivered vehicle is defective or not, and whether to give a compensation and the exact quantity.

3. BREACH AND REMEDIES

A contract can be breached when the debtor incurs in delay, fails to perform the service or executes it defectively. What would be more plausible is an improper compliance or unjust enrichment. However, **breach can still happen**, though smart contracts are created specially to avoid it, when a code cannot execute due to a missing input: for example, the executed code is meant to transfer an amount of cryptocurrencies but it is not deposited on a special account. Also, the **smart contract can be null when it produces an illegal outcome**, like selling drugs or allowing a minor to buy alcohol. To mitigate these problems some actions can be taken:

- Not only use smart contracts, but also a **written contract that has influence over the first** one and, thus, minimize the discrepancies.
- Write **computer codes in a precise way**, including variables that can adjust with the law and its changes. Parties are more likely going to set the terms according to the current law and terms that can be accommodated to future changes.
- Breach is a serious matter in Contract Law and how it will take place and outcomes are complicated to entirely predict. Then, it is worth to **encourage legislators and jurists to intervene in clarifying the consequences *ex-ante***. They can be interested in the enforcement of smart contracts because the intent of the parties is explicitly contained, and courts will have more certainty about it. For instance, specific matters in smart contracts can be expressly forbidden (such as drugs), demand permission and certain requirements to formalize a smart contract (when selling products of considerable value to not defraud tax

payment, block the deposited amount in a special account, etc.), implement a recognition system that detects the violation of laws (like when the interest of a loan become usury) or demands identification to prove legitimacy to contract (like not being a minor buying alcohol).

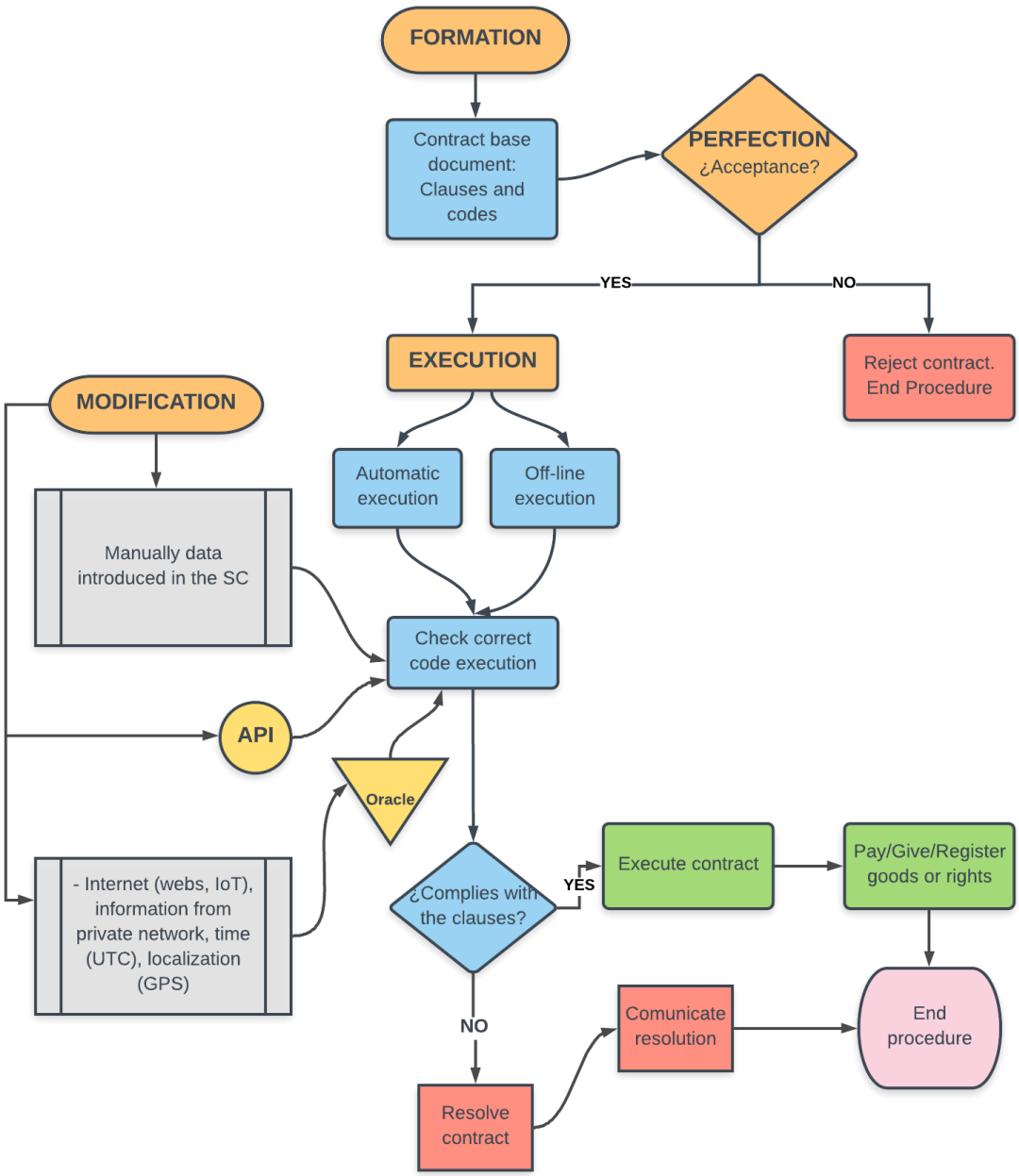
- Also, ***ex-post* remedies** can be a solution when regulation *ex ante* is not enough, demand compensation for the caused damages.

Whether *ex-ante* or *ex-post*, through regulation or legal action, there do not seem to be many differences to the ones applied in conventional contracts. What can be said is that in smart contract parties cannot rely on not knowing the terms defined and inserted, because the precision of these will determine the foreseeable outcome.

4. SMART CONTRACT DIAGRAM

A smart contract goes through different phases and in each of them it faces particularities that can be difficult to assume at a first glance. Therefore, in order to make it more comprehensible, here we offer a basic diagram (Figure 4) from beginning to the end of a smart contract, whether the execution was accomplished correctly or the procedure ends by resolution.

FIGURE 4: PHASES OF A SMART CONTRACT



Source: Own creation

V. CONCLUSIONS

A *smart contract*, in computer language, is not *per se* a contract but can be an agreement moulded by codes and data in the form of a software. It is considered under a legal sense when it complies with the requirements of the law regarding essential elements. Focusing on the second one, it is a digital contract deployed on the blockchain that encodes the terms and conditions of the parties that will automatically be executed once the determined events are triggered.

The perfection of a smart contract is produced with the consent by the concurrence of the offer and acceptance of the object and cause. That settled, there is no change of mind because the computer program is created to be self-enforceable. For parties not to argue not understanding what is being recorded, it is worth to be sure about: 1) Using a smart contract, with all its benefits and disadvantages; 2) Translating into code in a very clear and specific manner the intentions, and the possibility of withholding performance; 3) Using just a conventional contract or one that agrees on a smart contract but also regulates sensitive matters, like consent or obligations of means, and not leaving it all on the smart contract.

However, the complexity of reality obliges smart contracts to adapt. Thus, it is worth to consider modification of only certain aspects -leaving the rest immutable- by the parties relying on themselves or public authorities with an Application Programming Interface (API), as well as an oracle to update the smart contract.

As for what has been argued, a smart contract complies with the essential elements -consent, object and cause- and form Spanish Law requires and, therefore, fits under the principals of Contract Law. Nevertheless, it is still a new technology and we have a lot to learn. That is why it is worth encouraging public authorities and jurists to embrace and adapt to the changes and needs of the blockchain technology and smart contracts, taking into account its particularities, which are not few, in the stages of formation and perfection, performance and breach.

Most solutions come in comparison to traditional contracts, but it is not always possible due to the nature of smart contracts. Yet any type of contract is an agreement

that can be enforced, there are differences regarding the type of contract. In a conventional contract, when breached, the aggrieved party takes legal action going to court to demand restitution, a specific performance or pay for caused damages; But when it comes to smart contracts, due to its conditional and self-enforceability nature, the contract would have already been executed or in the process of execution, so the aggrieved party will have to go to court after an improper compliance or unjust enrichment. Nevertheless, the breach is a possibility when a code cannot correctly execute due to a missing input, yet smart contracts pretend to avoid these problems. For legal security, it is important that legislators and jurists intervene clarifying the consequences *ex-ante*.

It is true that blockchain and smart contracts pretend to change the way we use contracts. Now, many companies and governments[48] are working on these technologies due to all the benefits it provides, like lowering costs, being more secure, faster and, of course, with certainty. However, the main downsides to consider is related to the early stage of development and, in particular, if they can work with the existing laws or need an additional regulation. For now, the applications will not grab the attention of individuals and be focused on specific business areas, such as banking and insurance. We do not believe smart contracts will replace conventional contracts. It is an alternative in specific areas that provides considerable advantages.

[48] Turula, T. (2017). *Sweden is Trialling a Blockchain-Powered Land Registry*. [online] Nordic.businessinsider.com. Available at: https://nordic.businessinsider.com/sweden-is-pioneering-a-blockchain-run-land-registry---which-could-save-taxpayers-$100-million-2017-4/ [Accessed 9 Apr. 2018].

VI. BIBLIOGRAPHY

A) *Books and Papers*

1. Asharaf, S. and Adarsh, S. (2017). *Decentralized computing using blockchain technologies and smart contracts*. Hershey: IGI Global.

2. Bashir, I. (n.d.). *Mastering blockchain*. Packt.

3. Catchlove, P. (2017). *Smart Contracts: A New Era of Contract Use*. [online] Queensland University of Technology. Available at: https://ssrn.com/abstract=3090226 [Accessed 31 May 2018].

4. Chandler, R. (2016). *Smart contracts*. Wroclaw: Amazon Fulfilment.

5. Cong, L. and He, Z. (2018). *Blockchain Disruption and Smart Contracts*. [online]. Available at: https://ssrn.com/abstract=2985764 [Accessed 4 Jun. 2018].

6. Cong, L. and He, Z. (2018). *Blockchain Disruption and Smart Contracts*. [online] Available at: https://ssrn.com/abstract=2985764 [Accessed 15 Jul 2018].

7. Crosby, M., Pattanayak, P., Verma, S. and Kalyanaraman, V. (2015). *Blockchain Technology Beyond Bitcoin*. [online] Berkeley: Sutarja Center. Available at: http://scet.berkeley.edu/wp-content/uploads/BlockchainPaper.pdf [Accessed 13 Apr. 2018].

8. Deloitte (2018). *What is Blockchain?* [online]. Available at: https://www2.deloitte.com/content/dam/Deloitte/uk/Documents/Innovation/deloitte-uk-what-is-blockchain-2016.pdf [Accessed 4 Apr. 2018].

9. Dolader Retamal, C., Bel Roig, J. and Muñoz Tapia J. (n.d.). *La Blockchain: Fundamentos, aplicaciones y relación con otras tecnologías disruptivas*. [online] Cataluña: Universitat Politécnica de Catalunya. Available at: http://www.mincotur.gob.es/Publicaciones/Publicacionesperiodicas/EconomiaIndustrial/RevistaEconomiaIndustrial/405/DOLADER,%20BEL%20Y%20MUÑOZ.pdf [Accessed 16 Feb. 2018].

10. Drescher, D. (2017). *Blockchain Basics: A Non-Technical Introduction in 25 Steps*.

11. EquiSoft (2017). *La cadena de bloques (blockchain) Una tecnología disruptiva con el poder de revolucionar el sector financiero*. [online]. Available at: https://www.equisoft.com/wp-content/uploads/2017/09/White-paper-Blockchain-ESP-1.pdf [Accessed 8 Mar. 2018].

12. European Parliamentary Research Service (2017). *How blockchain technology could change our lives*. [online] STOA. Available at:

http://www.europarl.europa.eu/RegData/etudes/IDAN/2017/581948/EPRS_IDA(2017)581948_EN.pdf [Accessed 16 Feb. 2018].

13. Feliu Rey, J. (2018). *Smart Contract: Concepto, ecosistema y principales cuestiones del Derecho privado*. In: La Ley mercantil: Contratación mercantil, 47th ed. Wolters Kluwer.

14. Hazard, J. and Haapio, H. (2017). *Wise Contracts: Smart Contracts that Work for People and Machines*. [online] Available at: https://ssrn.com/abstract=2925871 [Accessed 15 Apr. 2018].

15. Latimer, P. (2011). *Signatures, Squiggles and Electronic Signatures*. [online] Swinburne University of Technology. Available at: https://ssrn.com/abstract=1601169 [Accessed 29 May 2018].

16. Lewis, A. (n.d.). *A Gentle Introduction To Blockchain Technology*. Gentle Introduction Reference Papers. [online] BraveNewCoin. Available at: https://bravenewcoin.com/assets/Reference-Papers/A-Gentle-Introduction/A-Gentle-Introduction-To-Blockchain-Technology-WEB.pdf [Accessed 16 Feb. 2018].

17. Mik, E. (2017). *Smart Contracts: Terminology, Technical Limitations and Real World Complexity*. [online] Singapore Management University. Available at: https://ssrn.com/abstract=3038406 [Accessed 11 Apr. 2018].

18. Mukhopadhyay, M. (2018). *Ethereum Smart Contract Development*. Birmingham: Packt Publishing.

19. Nakamoto, S. (2008). *Bitcoin: A Peer-to-Peer Electronic Cash System*. [online]. Available at: https://bitcoin.org/bitcoin.pdf [Accessed 4 Feb. 2018].

20. Navas Navarro, S., Robert, S., Górriz, C., Castells i Marquès, M., Camacho, S. and Mateo, I. (2017). *Inteligencia artificial*. Valencia: Tirant lo blanch.

21. Raskin, M. (2017). *The Law and Legality of Smart Contracts*. 1 Georgetown Law Technology Review 304. [online] Available at: https://ssrn.com/abstract=2959166 [Accessed 15 Apr. 2018].

22. Savelyev, A. (2016). *Contract Law 2.0: «Smart» Contracts as the Beginning of the End of Classic Contract Law*. [online] National Research University Higher School of Economics. Available at: https://ssrn.com/abstract=2885241 [Accessed 10 Mar. 2018].

23. Staples, M., Chen, S., Falamaki, S., Ponomarev, A., Rimba, P., Tran, A., Weber, I., Xu, X. and Zhu, J. (2017). *Risks and Opportunities for Systems Using Blockchain and Smart Contracts*. [online] Data61 (CSIRO). Available at:

https://publications.csiro.au/rpr/download?pid=csiro:EP175103&dsid=DS2 [Accessed 31 May 2018].

24. Tapscott, D. and Tapscott, A. (2017). *Blockchain Revolution: How the Technology Behind Bitcoin Is Changing Money, Business, and the World*. New York: Penguin.

25. Tasca, P., Aste, T., Pelizzon, L. and Perony, N. (2016). *Banking beyond banks and money: a guide to banking services in the twenty-first century*. Zurich: Springer.

26. Thompson, S. (2017). *The preservation of digital signatures on the blockchain*. [online] University of British Columbia. Available at: http://ojs.library.ubc.ca/index.php/seealso/article/view/188841/186525 [Accessed 31 Jun. 2018].

27. Tjong Tjin Tai, E. (2017). *Formalizing Contract Law for Smart Contracts*. Tilburg Private Law Working Paper Series No. 6/2017. [online] Tilburg Law School. Available at: https://ssrn.com/abstract=3038800 [Accessed 3 Jul. 2018].

28. Tur Faúndez, C. (2018). *Smart contracts*. 1st ed. Madrid: Reus.

29. Wattenhofer, R. (2016). *The science of the blockchain*. 1st ed. Inverted Forest.

B) ***Other***

30. Blockchain.info. (2018). *Miners Revenue*. [online] Available at: https://blockchain.info/charts/miners-revenue [Accessed 8 Apr. 2018].

31. Etherscripter (2018). [online] Available at: www.etherscripter.com [Accessed 2 May 2018].

32. Friebe, T. (2017). *Is Blockchain Equal to Blockchain?*. [online] Medium. Available at: https://medium.com/blockchainspace/2-introduction-to-blockchain-technology-eed4f089ce5d [Accessed 16 Jun. 2018].

33. Grincalaitis, M. (2018). *Can a Smart Contract be Upgraded/Modified?* [online] Medium. Available at: https://medium.com/@merunasgrincalaitis/can-a-smart-contract-be-upgraded-modified-1393e9b507a [Accessed 4 Jul. 2018].

34. Hogan Lovells (2018). *Insurance and reinsurance in Italy*. [online] Available at: https://www.lexology.com/library/detail.aspx?g=cd00da30-190c-4828-83ca-5e3ba1d02082 [Accessed 13 Mar. 2018].

35. Mattereum. (2018). [online] Available at: https://www.mattereum.com [Accessed 13 May 2018].

36. McLaughlin, E. (2018). *How blockchain works: An infographic explainer*. [online] SearchCIO. Available at: https://searchcio.techtarget.com/feature/How-blockchain-works-An-infographic-explainer [Accessed 30 Apr. 2018].

37. Preukschat, A. (2018). *Los contratos inteligentes serán cada vez más complejos gracias al Blockchain - elEconomista.es*. [online] Eleconomista.es. Available at: http://www.eleconomista.es/economia/noticias/8312353/04/17/Los-contratos-inteligentes-seran-cada-vez-mas-complejos-gracias-al-Blockchain.html [Accessed 6 Feb. 2018].

38. Ream, J., Chu, Y., Schatsky, D. (2016). *Upgrading blockchains: Smart contract use cases in industry*. [Blog] *Deloitte Insights*. Available at: https://www2.deloitte.com/insights/us/en/focus/signals-for-strategists/using-blockchain-for-smart-contracts.html [Accessed 15 Mar. 2018].

39. Sikorski, J., Haughton, J. and Kraft, M. (2017). *Blockchain technology in the chemical industry: Machine-to-machine electricity market*. [online] ScienceDirect. Available at: https://www.sciencedirect.com/science/article/pii/S0306261917302672 [Accessed 8 Feb. 2018].

40. Singh, A. (2018). *What makes a blockchain network immutable?* [online] Quora. Available at: https://www.quora.com/What-makes-a-blockchain-network-immutable [Accessed 5 Jul. 2018].

41. Stark, J. (2016). *Making Sense of Blockchain Smart Contracts*. [Blog] *Coindesk*. Available at: https://www.coindesk.com/making-sense-smart-contracts/ [Accessed 7 May 2018].

42. Szabo, N. (1996). *Smart Contracts: Building Blocks for Digital Markets*. [online] Fon.hum.uva.nl. Available at: http://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/smart_contracts_2.html [Accessed 13 Apr. 2018].

43. Turula, T. (2017). *Sweden is Trialling a Blockchain-Powered Land Registry*. [online] Nordic.businessinsider.com. Available at: https://nordic.businessinsider.com/sweden-is-pioneering-a-blockchain-run-land-registry---which-could-save-taxpayers-$100-million-2017-4/ [Accessed 9 Apr. 2018].

44. Vega, G. (2018). *Santander, BBVA, Sabadell, Bankia, Iberdrola, Gas Natural y Cepsa crean la mayor 'blockchain' de España*. [online] EL PAÍS RETINA. Available at: https://retina.elpais.com/retina/2017/05/30/tendencias/1496145136_731555.html