FACULTY OF
LAW AND CRIMINOLOGY

From clause to code: Transforming contracts into smart contracts

A study on the (im)possibility of coding contractual clauses

Word count: 44.543

Jan Joos

Student number: 01407484

Supervisor: Prof. Dr. Eva Lievens

Commissary: Mr. Carl Vander Maelen

A dissertation submitted to Ghent University in partial fulfilment of the requirements for the degree of Master of laws

Academic year: 2018 – 2019

GHENT
UNIVERSITY

ACKNOWLEDGEMENTS

"Every accomplishment starts with the decision to try" – John F. Kennedy

Five years ago, a younger version of myself hesitantly decided to study law at Ghent University. During the past five years, I have matured significantly, both on a personal and professional level. The Faculty of Law taught me how to think like a legal practitioner, and how to adopt a decent sense of criticism. I was offered many opportunities, such as an experience abroad, for which I wish to express my utmost gratitude. I enjoyed the enriching, challenging and above all interesting times I spent at Ghent University. On a personal level, the past five years have shaped my personality.

This Master's dissertation forms the pinnacle of those five years of legal studies. I would like to thank my thesis supervisor, professor Eva LIEVENS, for supporting me with interesting literature, for providing useful feedback and above all for granting me the opportunity to explore the newly developed science on smart contracts. Her enthusiasm and advice were of great value to me.

The writing of this dissertation brought along various emotions. I would like to thank my girlfriend Cato and roommates Simon, Christoph and Eva for their relentless optimism. None of this would have been possible without the support of my parents, who gave me every opportunity I could think of. I would like to thank my father in particular, for proof-reading this dissertation. I would also like to thank Angelo and Wout for guiding me through the fascinating world of computer science.

Jan Joos

Ghent, 15 May 2019

ABSTRACT (ENGLISH AND DUTCH)

This dissertation aims to research the possibility of transforming conventional commercial contract clauses into smart contract clauses (i.e. self-executing contract clauses). Such research is executed in three steps. Firstly, the technical specifications and the benefits of blockchain ledgers are described. Also, smart contracts are defined.

Secondly, through an extensive assessment of the existing literature on 21st century contracting practices, an overview is given on the impact which technology has had on such contracting practices, and to what extent technology can support the development of smart contracts. We conclude that contracting practices have remained relatively unchanged for the last thirty years, as compared to the relentless developments of technology. Modern day technologies might contribute to the commercial implementation of smart contracts. In addition, we will present a new hybrid contract model which synthesizes natural language contracts, smart contracts and existing technologies. The contract model is intended to introduce a paradigm shift towards human- and machine-readable contracts.

Thirdly, we analyse the essential characteristics of computer programs and computer languages. Moreover, we compare natural language with code in terms of vagueness and ambiguity. Several solutions are proposed to quantify abstract legal terms, such as reasonableness. Finally, we integrate several commercial contract clauses into the proposed hybrid contract model. We can conclude that, with the exception of specific non-operational clauses such as confidentiality clauses, a great amount of clauses can be adopted in smart contract form (i.e. in self-executable form).

Deze Masterscriptie onderzoekt de mogelijkheid tot het vertalen van conventionele contractuele clausules naar "slimme" contractuele clausules *(smart contract clauses)* (i.e. zelf-uitvoerende contractuele clausules). Het onderzoek is opgesteld volgens drie stappen. In de eerste stap worden de voordelen en de technische specificaties van *blockchain* toegelicht. Vervolgens definiëren we *smart contracts*.

In de tweede stap geven we, op basis van een grondige literatuurstudie, een overzicht weer van de impact die technologie momenteel heeft gehad op de contractuele gebruiken van de 21ste eeuw. We concluderen dat de voorbije dertig jaar contractuele gebruiken vrijwel onveranderd bleven, in vergelijking met de onophoudelijke ontwikkeling van technologie. Verder bespreken we in welke mate technologie een rol kan spelen in de commerciële implementatie van *smart contracts*. Daarnaast stellen we een nieuw hybride contractmodel voor, dat geschreven contracten, *smart contracts* en technologie samenbrengt. Het contractmodel beoogt aan te zetten tot een wijziging in de algemene opvatting ten

aanzien van contracten. Het stelt daarom contracten voor die interpreteerbaar zijn voor mens en machine.

In de derde en laatste stap analyseren we de essentiële karakteristieken van computerprogramma's en programmeertaal. We vergelijken natuurlijke taal met code op vlak van vaagheid en ambiguïteit. Verschillende methoden worden voorgesteld om abstracte juridische termen, zoals "redelijkheid" te kwantificeren. Ten slotte integreren we verschillende types van commerciële contractsbepalingen in het hierboven omschreven hybride model. We concluderen dat, met uitzondering van bepaalde niet-operationele clausules, zoals een confidentialiteitsclausule, een groot aantal clausule kunnen opgenomen worden in *smart contract* vorm (i.e. in zelf-uitvoerbare vorm).

TABLE OF CONTENTS

TITLE I. INTRODUCTION **12**

TITLE II. SCOPE OF THE RESEARCH **15**

1.1. Research question 15
1.2. Relevance of the research 16
1.3. Research method 16

TITLE III. TECHNICAL AND LEGAL FRAMEWORK **19**

CHAPTER 1. Blockchain **19**

1.1. Technical specifications 20
a. Chain of blocks 20
b. Hash: a digital fingerprint 21
c. Distributed ledger 22
1.2. Disruptive character 23
a. Distributed trust 24
b. Transaction efficiency 25
c. Durability 25
d. Immutability 25

CHAPTER 2. Smart contracts **25**

2.1. Defining smart contracts 26
a. In abstracto 26
i. Smart contract code – Code is law 26
ii. Smart legal contract – Law is code 28
iii. Legal framework 29
iv. The necessity of a blockchain network 30
v. Synthesis 31
b. In concreto 32
2.2. Beyond Bitcoin 34
a. Turing completeness 35
b. Binary state 35
c. A next generation blockchain: new applications 36
2.3. Oracles 36

CHAPTER 3. Preliminary conclusion **38**

TITLE IV. A PARADIGM SHIFT: NEXT GENERATION CONTRACTS 39

CHAPTER 1. 21st century contracting practices and technology 40

1.1. Contract management software 40

1.2. Artificial intelligence integration 41

CHAPTER 2. A new contract model for human and machine 43

2.1. Desirability of a paradigm shift 43

2.2. Legal design – *User-centred contracts* 44

2.3. Smarter contracts – *Machine-centred contracts* 45

a. Nick Szabo's smart contracts 45

b. Smarter contracts 46

i. External smart contract 46

ii. Internal smart contract 47

2.4. Synthesis: Smart Ricardian contracts? 48

2.5. Contracts for machines and humans: a new contracting model 50

a. Input 50

i. Human input 50

ii. Machine input 51

b. Output 51

i. Human output 51

ii. Machine output 52

CHAPTER 3. Opportunities for smart contracts 53

3.1. Use cases 53

a. Financial transactions 53

b. Internet of Things and smart property 55

c. Intellectual property rights 55

d. E-voting and share/token exchange 56

e. Insurance 58

f. Supply chain 59

3.2. Disruption? 60

a. Legal challenges 61

i. Establishing capacity 61

ii. Formation 61

iii. Interpretation 62

iv. Performance and modification 62

b. Non-legal challenges 63

CHAPTER 4. Preliminary conclusion 65

TITLE V. DRAFTING SMART CONTRACTS 67

CHAPTER 1. Programming language 68
1.1. Computer program 68
a. A set of instructions 68
b. Statements 69
c. Expressions 70
d. Functions 70
1.2. Computer language 71
a. Types of code 71
i. Machine code – Binary 71
ii. Source code – Syntax and semantics 71
iii. Pseudo-code – Legible code 72
b. Types of languages 72

CHAPTER 2. Remedying vagueness 73
2.1. Defining vagueness 73
2.2. Elimination of vagueness 73
2.3. Semantic issues 75
a. Ambiguity 75
b. Translating abstract and non-quantifiable concepts to code 76
i. Quantification through machine learning 76
ii. Quantification through oracle input 77

CHAPTER 3. Mapping contract clauses 78
3.1. Time related clauses 78
a. Legal significance 78
i. Term clauses 78
ii. Time of performance 79
b. Coding possibilities 79
c. Integration in the new contract model 82
3.2. Financial clauses 83
a. Legal significance 83
i. Payment clauses 84
ii. Determination of price 84
iii. Escrow 85
b. Coding possibilities 85

c. Integration in the new contract model 88
3.3. Clauses related to physical objects 89
a. Legal significance 89
b. Coding possibilities 91
i. A sufficient level of accuracy 91
ii. A sufficient level of reliability 94
c. Integration in the new contract model 95
3.4. Boilerplate clauses 97
a. Confidentiality clause 97
b. Assignment clause 98
c. Impediment clause 99
i. First step: The cause of non-performance 100
ii. Second step: The attributability of the cause 101
d. Arbitration clause 102
e. Good faith (clause) 103

CHAPTER 4. Conclusion 106
a. Input 107
b. Output 107

TITLE VI. FINAL CONSIDERATIONS 109

TITLE VII. BIBLIOGRAPHY 110

TITLE I. INTRODUCTION

The future is already here — it's just not very evenly distributed

- William Gibson, *The Economist 2003*

1. In 1994, the American computer scientist and legal scholar Nick SZABO launched the idea of the smart contract. Back then, SZABO defined a smart contract as follows:

> *A smart contract is a computerized transaction protocol that executes the terms of a contract. The general objectives of smart contract design are to satisfy common contractual conditions (such as payment terms, liens, confidentiality, and even enforcement), minimize exceptions both malicious and accidental, and minimize the need for trusted intermediaries. Related economic goals include lowering fraud loss, arbitration and enforcement costs, and other transaction costs.*[1]

A few years later, SZABO elaborated and clarified the concept of smart contracts by the example of the vending machine.[2] The machine operates according to simple and previously determined rules. Upon entering coins amounting to a required value, the machine will provide a soda and, if necessary, the change. This simplified sales transaction demonstrates the logic behind the execution. By analogy, smart contracts will be executed just like the vending machine, when certain predetermined conditions are met. SZABO argued that many kinds of contractual clauses could be embedded in the hardware and software humans interact with.[3] As an execution-based tool, smart contracts could increase cost efficiency and guarantee contract execution. Parties can be reassured that one is going to get what is contracted.[4] Trust would become a matter of course, as neither contractual party would be capable of committing fraud or neglect the performance of the contract.

---

[1] N. SZABO, "Smart Contracts", 1994, http://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/smart.contracts.html (Consulted on 18 November 2018).

[2] N. SZABO, "Formalizing and Securing Relationships on Public Networks", *First Monday,* 1997, https://ojphi.org/ojs/index.php/fm/article/view/548/469 (Accessed 18 November 2018).

[3] *Ibid.*

[4] E. TJONG TJIN TAI, "Formalizing Contract Law for Smart Contracts", *Tilburg Private Law Working Paper Series*, 2017, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3038800 (Consulted on 16 November 2018).

2. Although SZABO realised the potential impact of smart contracts on businesses, the necessary technical infrastructure to deploy such contracts on a large scale was not yet available.[5] Conventional digital ledgers were prone to fraud and hacking. The emergence of Bitcoin in 2008 changed the technological landscape and created new opportunities.[6] Besides launching the aspirational concept of virtual currencies, Bitcoin indirectly revealed the underlying opportunities of the distributed ledger technology upon which it was built.[7] These advanced forms of digital ledgers could guarantee the security of the data that was stored on the ledger.[8] As Bitcoin itself could only be a platform to support less complex transactions, new platforms arose.[9] Ethereum, Ripple and many other platforms have been using the distributed ledger technology to provide the opportunity to develop more intricate distributed applications. Smart contracts, or "multifunctional vending machines" finally found a fertile soil to blossom and to live up to the expectations.

3. Today, the term "smart contract" forms part of major technological developments and can be associated with other buzzwords, such as Bitcoin and blockchain. Even though a prosperous future is promised for these technologies, there has been and still is a substantial amount of scepticism by experts and legal practitioners.[10] One of the major points of criticism is related to the hypothetical character smart contracts still surrounds. No paradigm shift has been verified so far: contracts are still being drafted in writing, rather than in code. As these technological developments are on the brink of a major breakthrough, the majority of the literature still particularises the *potential* opportunities of smart contracts. Until now factual evidence of successful implementations is uncommon. According to the Gartner Hype Cycle[11], smart contracts are still on the rise to the peak of inflated expectations, whereas blockchain already passed the peak and entered the through of disillusionment.[12] In other words, the perception towards blockchain applications has become more pragmatic and down to earth. On the contrary, smart contracts are still pictured as a passe-partout solution for many issues, therefore being

---

[5] S. OMOHUNDRO, "Cryptocurrencies, smart contracts, and artificial intelligence", *AI Matters*, 2014, vol. 2, 19–21; M. ALHARBY en A. VAN MOORSEL, "Blockchain-based smart contracts: A systematic mapping study", *ArXiv Prepr. ArXiv171006372*, 2017, https://arxiv.org/abs/1710.06372 (Consulted on 18 November 2018).

[6] S. NAKAMOTO, "Bitcoin: A peer-to-peer electronic cash system", 2008, https://bitcoin.org/bitcoin.pdf (Consulted on 18 November 2018).

[7] V. BUTERIN, "A next-generation smart contract and decentralized application platform", *White Pap.*, 2014, https://www.ethereum.org/ pdfs/EthereumWhitePaper.pdf (Consulted on 18 November 2018).

[8] *Infra,* 14.

[9] *Infra,* 43.

[10] K. O'HARA, "Smart contracts - Dumb idea", *IEEE Internet Computing*, 2017, vol. 2, 97–101; M. GIANCASPRO, "Is a 'smart contract' really a smart idea? Insights from a legal perspective", *Computer Law & Security Review*, 2017, vol. 6, 825–835, http://www.sciencedirect.com/science/article/pii/S026736491730167X.

[11] The Gartner Hype Cycle is a statistic tool used to assess new technologies. It distinguishes the hype from the commercially viable technologies by presenting those technologies on a cycle, according to the expectations over time.

[12] X., "Hype cycle for blockchain technologies 2018", 2018, https://www.gartner.com/doc/3883991/hype-cycle-blockchain-technologies- (Consulted on op 18 November 2018).

overhyped. The enthusiasm to overpromote this technology might harm its long-term prospects.[13] It is therefore necessary to acknowledge that smart contracts still reside at a very early stage of commercial viability.

In general, market forces serve as the engine for technological development. Potential profitability will encourage businesses to adopt and develop these new technologies. For example, when the automobile was brought to the market in the beginning of the 20th century by Ford, other businesses were stimulated to adapt their business plan and outcompete competitors by being innovative. Such competition stimulates progress in technology. On that moment, no legislation on car traffic was adopted. It was deemed necessary when a large group of citizens had purchased a car, and traffic conflicts arose. The legislator therefore only interferes after these technological developments are externalised in business and society and cause further conflicts with the existing boundaries of the law.

By consequence, little or no legal framework has yet been settled to integrate smart contracts. Several authors predict an extensive amount of complications with respect to the existing legislation.[14] Matters such as establishing capacity, contracting under mistake and methods of interpretation might not correspond with the legal characteristics of smart contracts in accordance with the current law of contracts.[15] Smart contracts undoubtedly challenge the legal order, providing an interesting issue for further legal discussion.

4. Smart contracts resort at the crossroads of law and computer science. In essence, the contract executes itself in accordance with its previously determined coded clauses. The legal capacities of a smart contract are limited by the technical capacities necessary to create these coded contractual clauses. Therefore, not only the limitations arising from the current legal order should be observed, but also whether computer science establishes limits to the contracting capacities of parties.

---

[13] C. MULLIGAN, J.Z. SCOTT, S. WARREN en J. RANGASWAMI, "Blockchain beyond the hype; a practical framework for business leaders", *white paper of the World Economic Forum*, 2018, http://www3.weforum.org/docs/48423_Whether_Blockchain_WP.pdf (Consulted on 4 March 2019).

[14] M. RASKIN, "The Law and Legality of Smart Contracts", *Georgetown Law Technology Review*, 2017, vol. 2, 321–328; M. GIANCASPRO, "Is a 'smart contract' really a smart idea? Insights from a legal perspective", *supra* from 10, 2017, vol. 6, 828–833; E. TJONG TJIN TAI, "Formalizing Contract Law for Smart Contracts", *Tilburg Private Law Working Paper Series*, 2017, 3-4, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3038800 (Consulted on 16 November 2018).

[15] M. GIANCASPRO, "Is a 'smart contract' really a smart idea? Insights from a legal perspective", *supra* from 10, 2017, vol. 6, 825–835.

TITLE II. SCOPE OF THE RESEARCH

1.1. Research question

5. In order to prevent this research from becoming superficial and imprecise, it is necessary to define its limitations in advance. Hence, it should be noted that this research does not pursue to focus on every aspect of smart contracts. As mentioned before, the concept of smart contracts leads to multiple questions which, for the time being, will remain unanswered. Therefore, it is of great importance to limit the scope of this research. This study will not deal with any questions raised with regard to the validity of smart contracts, nor their place within the actual law of obligations. A lot of scientific research already has been conducted on these matters. Evidently, the viability of smart contracts is based on their legality in contract law. However, this study will not attempt to provide an answer to these questions. It will only focus on the content of the contracts, rather than on the formality and the validity. Moreover, not every potential contract clause can be analysed in this study. A more realistic approach would require a limited amount of clauses to be touched upon. The relevant clauses will be picked according to the type of contract clauses that are presumably useful in a smart contract context. We will mainly emphasize on commercial contracts. Additionally, this study will mainly focus on rather standard clauses, as opposed to more complex and specific clauses. Moreover, due to this study's interdisciplinary character, full technical details of each coding process will fall outside of the scope of this master's thesis. The research will be executed from a judicial point of view, by examining the requirements, limits and benefits of the law for smart contracts, as opposed to the limits of code and technology.

6. In the light of the foregoing, a central research question can be expressed. This research intends to provide an answer to the question whether traditional (commercial) contracts can be translated to smart contracts. Before a sophisticated answer to such a question can be expressed, several aspects and sub-questions need to be examined first, such as the definition and the conditions for the existence of smart contracts. Moreover, this research will attempt to answer several sub-questions:

- What benefits does a blockchain network bring for the development of smart contracts?
- What are the most promising use cases for smart contracts?
- What are the existing initiatives to incorporate technological developments into contracting practices?
- Which language is the most adequate to express contract clauses in a coded version?
- Will courts be capable of interpreting smart contracts?
- Which functions are necessary to express clauses in code language?

- How can abstract concepts, such as reasonableness, be adopted in smart contracts?
- Are certain clauses impossible to express in code language?
- What level of expertise is required to write an operational smart contract as a lawyer?

1.2. Relevance of the research

7. All academic research must pursue a dual importance: scientific relevance and social relevance. With regard to social relevance, a comprehensive system to bridge the gap between a contractual clause and a coded clause may be of great use for legal entrepreneurs and lawyers. There is still little or no systematic approach towards the construction of smart contracts. If parties decide to arrange their contractual relationship by means of a smart contract, there is no solution to reduce the high cost of creating one. Few businesses are capable of constructing smart contracts and anchor them into the blockchain in a cost-efficient and lucrative way. Lowering the entrance costs for businesses interested in the market of smart contracts would enhance accessibility and competition.
With regard to the scientific relevance, the research aims to achieve new knowledge on the one hand and to synthesize the existing knowledge that is currently scattered over multiple resources on the other hand. No comprehensive work on the translation of written contractual clauses has been conducted yet. Moreover, new knowledge will be generated by applying several research methods to existing types of clauses.

1.3. Research method

8. This research departs from a technical perspective through an extensive study of the existing literature on the subject of blockchain and smart contracts. International resources are consulted, due to the universal and overarching character of the aforementioned phenomena. As this research was partially conducted during an exchange with the University of Glasgow, a wide range of additional literature was available to benefit the comprehensiveness of this study. Due to the very early stage in which the smart contract technology currently resides, there are few possibilities of performing relevant case studies. Nonetheless, we attempt to provide some examples of initiated smart contract applications. The interdisciplinary character of this research implies the necessity of consulting sources outside legal spheres, focussing more on a computer science related aspect. To support such interdisciplinary approach, several interviews are conducted with professionals and academics involved in computer science affairs.

9. Generally, this research comprises three parts, which are adopted in Title III, IV and V. In the first part, or Title III, the technical specifications of blockchain technology and smart contracts are described. We outline the benefits which are created by integrating a blockchain based network. Moreover, we attempt to define smart contracts, based on the existing literature and various opinions.

This establishes an overview of the phenomenon of smart contracts from a technical and legal perspective. In doing so, part one, or Title III, attempt to answer the research sub-questions relating to the role of blockchain and the definition of smart contracts. The research for this Title is predominantly based on the existing literature on cryptography, decentralised ledgers, blockchain and smart contracts. Also, information from the conducted interviews is used to add insights from a practical perspective.

10. Subsequently, the second part, or Title IV, describes the current contracting practices and how they relate and adapt to new technologies. Modern day contracting practises are illustrated in the light of potential paradigm shifts.[16] The second part presents a new hybrid contract model, integrating several aspects of these modern developments. The model attempts to integrate smart contracts into the existing environment of natural language contracts. This contract model acts as a synthesis of the second part and is used in the third part to assess whether contract clauses can be transformed into coded clauses. Moreover, we provide a summary of existing smart contract initiatives and opportunities, based on a selection of case studies.

11. The third and final part, or Title V, examines the possibilities with regard to the transformation of traditional contract clauses to coded clauses. A brief introduction in the logic behind computer programs and programming languages is given. This overview is based on introductory handbooks on programming, weekly seminars on computer science (Harvard CS50 online education) and discussions with computer scientists and professional programmers. Also, we compare natural language clauses with code in terms of vagueness and ambiguity. Several methods to quantify abstract legal concepts, such as reasonableness, are presented. Subsequently, we apply this theoretical framework to contract clauses to assess whether or not a translation to code is possible. Depending on the outcome of such assessment, the clauses are integrated in the hybrid contract model, as explained in the second part.
For the purpose of this research, general principles of contract law function as standards to determine the structure and legality of certain clauses.[17] Such a choice is made to maintain a high-level and universal perspective on both sides of the comparison between code and language: from a programmer's perspective, pseudo-code principles are used to explain the mechanisms of code; from a legal perspective, the Draft Common Frame of Reference and the Principles of European Contract Law are used to assess the ability to code contract clauses, rather than using national legislation as a benchmark instead.[18] As mentioned before, a selective approach will be maintained by analysing those clauses

---

[16] Paradigm shifts refer to fundamental changes in the perception of things.
[17] *Infra*, 129.
[18] C. VON BAR, E. CLIVE en H. SCHULTE-NÖLKE, *Principles, definitions and model rules of European private law: draft common frame of reference (DCFR)*, Oxford, Oxford University Press, 2009; O. LANDO en H.G. BEALE, *Principles of European contract law: Parts I and II*, The Hague, Kluwer Law International BV, 2000.

which can be relevant for smart contract opportunities, as described in Title IV. We will focus mainly on commercial contracts. In doing so, we will attempt to answer the central research question.

TITLE III. TECHNICAL AND LEGAL FRAMEWORK

12. In order to understand the opportunities and limitations of smart contracts, a holistic insight into the technical and legal nature is essential. This Title will provide an outline of the most relevant technical concepts involved in the deployment of smart contract technology. In the light of the legal perspective this research paper will apply, technical complexity will be limited to the amount necessary to understand the following titles.

The development of smart contracts in the literature is associated with the rise of three development stages of blockchain or so-called 'blockchain generations'.[19] The first generation refers to virtual currencies, such as Bitcoin. The second generation are smart contracts. The third and final generation are decentralised autonomous organisations, which will be discussed later in this research. Therefore, we will first describe blockchain, its role in supporting smart contract development and its potential disruptive impact. Subsequently, smart contracts will be discussed in terms of their definition and their technical specifications. Several types of definitions will be compared. Based on these discussed definitions, a synthesis definition for this research will be expressed. Finally, we will describe the role of oracles, as they bridge the gap between the online environment of smart contracts and the physical world.

CHAPTER 1. BLOCKCHAIN

13. Blockchain has been granted the status of the next revolutionary, or disruptive, technology.[20] SWAN describes it as "fundamental for forward progress in society as Magna Charta or the Rosetta Stone".[21] Its popularity emerged in conjunction with the rise of Bitcoin. Creating the first decentralised cryptocurrency, Satoshi NAKAMOTO introduced Bitcoin to the world in 2009.[22] Bitcoin was originally intended as a peer-to-peer version of electronic cash to allow payments being transferred from one party to another.[23] Crucial was the fact that parties no longer had to rely on traditional financial institutions to complete their financial transactions. This also implied a significant legal vacuum, as such transactions were not regulated at all. Bitcoin gained people's interest on a massive scale and turned

---

[19] M. DEMEYER, *Blockchain Technology And Smart Contracts From A Financial Law Perspective*, Ghent, University of Ghent, 2018, 13, https://lib.ugent.be/fulltxt/RUG01/002/479/332/RUG01-002479332_2018_0001_AC.pdf (Consulted on 13 March 2019).

[20] C. BERG, S. DAVIDSON en J. POTTS, "The blockchain revolution", *The Institute of Public Affairs Review: A Quarterly Review of Politics and Public Affairs*, 2017, vol. 4, 35.

[21] M. SWAN, *Blockchain: Blueprint for a new economy*, O'Reilly Media, Inc., 2015, vii.

[22] S. NAKAMOTO, "Bitcoin: A peer-to-peer electronic cash system", *supra* from 6, 2008.

[23] *Ibid.*

out to become more of a speculative investment, causing high volatility.[24] Apart from demonstrating the substantial consequences of the entrance of cryptocurrencies in financial markets, Bitcoin established the belief in the commercial viability of the technology upon which it was built: blockchain.

1.1. Technical specifications

a. Chain of blocks

14. A blockchain is a digital concept to store data chronologically and publicly through the internet. Under the form of "blocks", data is added to the already existing chain of blocks, hence the name blockchain.[25] DRESCHER draws the analogy between blockchain and the pages in a book. Each page contains words and sentences and is chained to the previous page. As the words are written on different pages, instead of being written on a large spool, the pages can be considered "blocks", and the book a "blockchain".[26] Equally to the pages in a book, it is impossible to chronologically alter or remove blocks in the chain once they are established. The data represented by the blocks can be anything: not only money but also value in general or a transaction. Certain blockchain systems impose restrictions on the types of data that can be stored.[27]

This chain of blocks is embedded in an online environment. As the data contained in the blocks can be used to determine owners of digital assets[28], it is subject to the risk of double spending.[29] This risk refers to the fact that digital goods can be copied without perceptible restrictions and to the situation where the ownership of an asset is transferred more than once. Upon manifestation of the risk, the integrity of the blockchain might be affected, as the same transaction might be executed twice. The information which is consolidated in the blocks can no longer be perceived as authentic. Owners of digital assets might lose their entitlement.[30]

15. Bitcoin faced these challenges and managed to overcome the double spending conundrum. There are several mechanisms that maintain the integrity of the blockchain network. A first mechanism

---

[24] Volatility refers to the fluctuation in price of a certain asset over time; B.M. BLAU, "Price dynamics and speculative trading in bitcoin", *Research in International Business and Finance*, 2017, vol. 41, 15.
[25] D. DRESCHER, *Blockchain Basics: A Non-Technical Introduction in 25 Steps*, New York, Springer, 2017, 34.
[26] *Ibid.*
[27] The blockchain upon which Bitcoin is built is not designed to support smart contract protocols.
[28] Digital assets contain among others: cryptocurrencies, crypto commodities, utility tokens, security tokens and real-world asset tokens. For more details on digital assets, see A. BACK, "Crypto Assets: Beyond Cryptocurrencies to a New Digital Asset Class", *The Blockchain Review*, 2018, https://blockchainreview.io/crypto-assets-cryptocurrencies-digital-blockchain-stablecoin-sto-bitcoin-altcoin/ (Consulted on 27 November 2018).
[29] D. DRESCHER, *Blockchain Basics: A Non-Technical Introduction in 25 Steps*, *supra* from 24, 2017, 51; B. SINGHAL, G. DHAMEJA en P.S. PANDA, "How blockchain works" in *Beginning Blockchain: A Beginner's Guide to Building Blockchain Solutions*, Springer, 2018, 126.
[30] The property of Bitcoins can (according to Belgian law) be seen as a claim. For more details, see S. GEIREGAT, "Eigendom op bitcoins", *RW*, 2017, vol. 27, 1043–1050.

is based on cryptography and is used to create a digital fingerprint (i.e. a "hash") for each block. The hash is used on the one hand to create a chronological structure between the individual blocks, and on the other hand to protect the system against tampering. The second mechanism refers to the decentralised aspect of a blockchain-based ledger: there is no single entity charged with the management of the ledger. Such decentralisation provides an even greater safeguard against malicious behaviour, guaranteeing an immutable ledger.

b. Hash: a digital fingerprint

16. Firstly, the individual blocks in the chain contain, apart from data (i.e. transaction data or value), a 'hash'. This hash is the so-called digital fingerprint of a block: it aims to allocate a unique identifier to each block. It consists of a fixed amount of numbers and letters. The purpose of such 'fingerprint' is to be able to compare data without comparing the explicit content of the data.[31] This hash (i.e. a specific pattern of characters allocated to each block) allows to do so. Changes to the data in the block imply changes to the hash as well, providing a system to detect variations in the data.

In addition, every block contains the hash of the previous block.[32] By combining the allocation of a unique identifier with the adoption of the hash of the previous block, tamper with an anterior block will result in the invalidity of the following block that includes the former's hash. A chain reaction will occur, invalidating all following blocks in the blockchain. This is the first step in making the blockchain tamper-proof: malicious parties would have to tamper with one block while having to re-calculate and adjust all the hashes of the following blocks in order to maintain the integrity of the blockchain. However, modern-day computer hardware has potentially the ability to calculate the hashes of all the following blocks to manipulate the blockchain successfully. Bitcoin acknowledged this possibility and anticipated such manipulation by establishing a proof-of-work mechanism, which will be explained later.[33] In summary, this mechanism aims to prolong the time necessary to calculate these hashes, if an attacker would attempt to threaten the integrity of the system. The hacker should calculate a so-called "nonce" to find a block's hash. In the case of Bitcoin, the proof-of-work[34] requires approximately ten minutes for each block to calculate the nonce.[35, 36] In other words, for every block the hacker wants to

---

[31] D. DRESCHER, *Blockchain Basics: A Non-Technical Introduction in 25 Steps*, *supra* from 24, 2017, 81-92.
[32] B. SINGHAL, G. DHAMEJA en P.S. PANDA, "How blockchain works" in *Beginning Blockchain: A Beginner's Guide to Building Blockchain Solutions*, Springer, 2018, 116-120.
[33] N. SHI, "A new proof-of-work mechanism for bitcoin", *Financial Innovation*, 2016, vol. 1, 1-2.
[34] *Supra*, 18
[35] *Ibid.*
[36] Ethereum is currently investigating another mechanism to solve the same problem, called the proof-of-stake. For more details, see B. BAMBROUGH, "Ethereum Price Jumps On Major Bank Approval And Approaching Proof-Of-Stake", *Forbes Magazine,* 2018, https://www.forbes.com/sites/billybambrough/2018/11/05/ethereum-price-jumps-on-major-bank-approval-and-approaching-proof-of-stake/#79b45db55621 (Consulted on op 27 November 2018).

manipulate, it would take at least 10 minutes to determine the right hash. Since every block needs to be manipulated to successfully hack the system, it is nearly impossible to attack the blockchain.

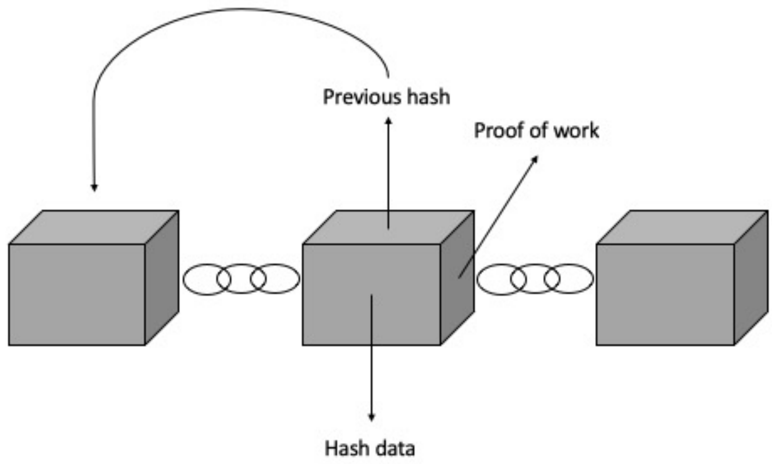


Figure 1 - Blockchain parts. In this visual representation of a block in the blockchain, three parts can be distinguished: the hash of the data embedded in the second block, the hash of the previous block and the key to solve the proof of work (the nonce).

c. Distributed ledger

17. Secondly, perhaps the most important element in securing the integrity of the blockchain is the existence of a distributed ledger. For ages, ledgers have been indispensable to record business transactions and identify owners, forming the backbone of economies. Anytime a consensus about certain facts is necessary, ledgers are used.[37] Evolution encompassed the rise and decline of a variety of systems, from physical ledgers to digital ledgers. Blockchain demarcates a new era. What distinguishes blockchain from an online database as we know it, is that it is a decentralised system. Blockchain no longer relies on a trusted central authority to maintain and validate the ledger.[38] Instead, all transactions are recorded in a shared ledger. This shared ledger is distributed among the user participants of the blockchain network, providing each of the users an identical and authentic version. These versions are updated simultaneously in accordance with new transactions added to the blockchain. Important to notice is the inexistence of one authentic version of the ledger: each participant in the network has an authentic version. A blockchain ledger therefore differs from cloud-based ledger systems that synchronize the state of the ledger on the different participating devices in the network.

18. The different participants in the network are called "nodes". The nodes in this peer-to-peer network do not need to trust each other, as they only need to trust the mechanism to reach a consensus

[37] C. BERG e.a., "The blockchain revolution", *supra* from 19, 2017, vol. 4, 35-36.
[38] *Ibid.*

that determines the validity of the blocks, which contain data on transactions. This mechanism does not necessarily rely on voting rights or negotiations between the nodes. A specific "lottery" mechanism determines the node that obtains the authority to approve a new block, thereby consolidating data concerning new transactions.[39] In the Bitcoin blockchain, the mechanism implies the validation of a new block by the node succeeding in solving the proof of work, or in other words, finding the nonce.[40] This cryptographic puzzle requires a significant amount of computational power to solve. Therefore, it becomes practically impossible for a node to solve the proof of work of consecutive blocks, as this would require more than 50% of the combined computational power held by the nodes together.[41] As Bitcoin transactions are broadcasted to every node, the nodes collect them into a block. The node solving the proof-of-work will broadcast the block to the other nodes, upon which they accept the validity of the transactions. Apart from containing Bitcoin transactions, the validated block also contains one Bitcoin, which is transacted to the node that found the proof of work. In doing so, Bitcoin nodes are encouraged to "mine" Bitcoins, or to acquire the additional (part of a) Bitcoin embedded in the new block by devoting their computational power. Aforementioned lottery mechanism can be structured through other mechanisms as well, such as the proof-of-stake.[42]

1.2. Disruptive character

19. Henry Ford once mentioned that if he had asked people what they wanted, they would have said 'faster horses'. The automobile is, just like the internet, an example of a disruptive technology. They emerge a new technological landscape (e.g. cars) which renders outpaced technologies obsolete (e.g. horseback riding). Over the last years, blockchain was hyped as such a disruptive technology, outdating ordinary (digital) databases and ledgers.[43] Insiders predicted a revolution in a number of industries such as financial transactions and supply chains.[44] However, in order to assess the actual impact of blockchain on modern society and to refrain from exaggerating its scope of application, it is crucial to understand the benefits of a blockchain system.

---

[39] As pointed out by B. PRENEEL during his lecture "Blockchain 101 – Technische aspecten van blockchain", Leuven, 21 June 2018.
[40] S. NAKAMOTO, "Bitcoin: A peer-to-peer electronic cash system", *supra* from 6, 2008, 3.
[41] Such a risk is referred to as a 51%-hack. The majority of the system is able to find the nonce for consecutive blocks. Hence, the majority can alter the transactions embedded in the block and coordinate the following blocks to match the former block's data.
[42] I. BENTOV, A. GABIZON en A. MIZRAHI, "Cryptocurrencies without proof of work", Springer, 2016, 142–157.
[43] B. MARR, "Here Are 10 Industries Blockchain Is Likely To Disrupt", *Forbes Magazine*, 16 July 2018, https://www.forbes.com/sites/bernardmarr/2018/07/16/here-are-10-industries-blockchain-is-likely-to-disrupt/#1d27a9c0b5a2 (Consulted on 24 December 2018).
[44] L. NOONAN, "Banks find a use for blockchain: cross border payments", *Financial Times*, 12 November 2018, https://www.ft.com/content/57b1064a-c1a5-11e8-84cd-9e601db069b8 (Consulted on 24 December 2018).

a. Distributed trust

20. Throughout the history of business, trust has been considered a precondition for transacting with a potential counterparty on the market. As a rational protective mechanism, parties adopted the custom of not trusting potential counterparties in advance. Instead, trust had to be earned by establishing a stable business relationship. This so-called trust gap was (partially) bridged by intermediaries, such as banks, credit reporting agencies, etc. In the situation where someone orders a book from Amazon, both buyer and seller of the book rely on Amazon as a central trusted third intermediary.

The ultimate value proposition of blockchain is the distribution of trust. A central trusted party is no longer necessary when transactions rely on distributed ledger technology, as trust is an inherent part of the network by default.[45] In other words, the blockchain network itself becomes the central trusted middle man. This is exemplified by Bitcoin. It excludes financial institutions from their role as middleman in financial transactions. Transaction fees are reduced because third parties are no longer involved, thereby increasing the speed, the efficiency and the security of the transaction.[46]

21. As a consequence, when there is no substantial advantage in eliminating the middle man, blockchain technology is not likely to fulfil its disruptive predetermination. A common misconception with regard to the relevance of blockchain are governmental facilities.[47] WÜST and GERVAIS point out that blockchain technology is only relevant when mutually mistrusted entities want to interact and change the state of a system (i.e. a ledger), and are not willing to agree on an online trusted third party.[48] Parties carrying out a financial transaction do not want to rely on strong trust assumptions.[49] Platforms such as Bitcoin and Ripple already demonstrated the potential for the financial sector. On the contrary, the marginal surplus value created through the integration of a distributed ledger in governmental affairs, such as land registry, should be rationalised. Unless state authorities are fraud-sensitive and corruption might be dominant, there is no intrinsic reason to doubt the trustworthiness of governments. A blockchain network might, however, constitute an adequate weapon to combat governmental corruption, as demonstrated in Georgia.[50] Nonetheless, a digital centralised land registry will be sufficient for more trustworthy governments.

---

[45] B. SINGHAL e.a., "How blockchain works", *supra* from 29, 2018, 23-24.
[46] K.D. WERBACH, "Trust, but verify: Why the blockchain needs the law", *Berkeley Technology Law Journal*, 2018, 488–550.
[47] As pointed out by B. PRENEEL during his lecture "Blockchain 101 – Technische aspecten van blockchain", Leuven, 21 June 2018.
[48] K. WÜST en A. GERVAIS, "Do you need a Blockchain?" in 2018 Crypto Valley Conference on Blockchain Technology (CVCBT), *IEEE*, 2018, 53.
[49] *Ibid.,* 51.
[50] N. KSHETRI, "Will blockchain emerge as a tool to break the poverty chain in the Global South?", *Third World Quarterly*, 2017, vol. 8, 1716.

b. Transaction efficiency

22. Transaction speed represents the rate at which data is transferred from one account to another. As a consequence of the aforementioned elimination of the middleman, fewer hurdles need to be crossed to complete a transaction. Through an integrated online network, nodes are capable of transacting instantaneously. However, large blockchain networks come with scalability challenges. Bitcoin was reported to support less than seven transactions per second. When compared to the ability of payment network Visa to provide a platform for 47,000 transactions per second, it is questionable whether Bitcoin will support a global payment system in the near future. Furthermore, nodes in the Bitcoin network would be confronted with an enormous amount of data attached to the blockchain.[51] Currently, research tries to overcome these scalability challenges.[52]

c. Durability

23. As opposed to an often limited amount of back-ups in a centralised network, the data embedded in the blockchain is by default saved and stored with every node in the network. Consequently, an enhanced safeguard against data loss is considered an advantage related to the usage of a blockchain network.

d. Immutability

24. As explained before, blockchain networks avert the possibility of tamper through cryptography.[53] Unwanted changes in the data are nearly unachievable. Blockchain networks might establish a more secure environment for sensitive data or digital assets.

CHAPTER 2. SMART CONTRACTS

25. The following chapter will describe the technical aspects of the next step in the blockchain evolutionary process. It comprises the emergence of so-called second-generation blockchain applications or smart contracts, shaping a new dimension of possibilities. At present, there's no general consensus yet on the definition of smart contracts. This chapter will describe several opinions and conflicting perceptions discussed in the literature. Eventually, we will attempt to express a synthesis definition. Nonetheless, while the rise of Bitcoin aroused a great interest for smart contracts, one could wonder whether or not the underlying platform should be a decentralised ledger to confirm the existence

---

J. POON en T. DRYJA, "The bitcoin lightning network: Scalable off-chain instant payments", 2016, 2, https://www.bitcoinlightning.com/wp-content/uploads/2018/03/lightning-network-paper.pdf

[52] For more, see E.S. THORSRUD, *Long-term Bitcoin Scalability*, Thesis, Trondheim, Norwegian University of Science and Technology (NTNU), 2018.

[53] *Supra*, 16.

of a smart contract. In other words, is a decentralised ledger a *conditio sine qua non* to create smart contracts? We will attempt to provide an answer to such question.

2.1. Defining smart contracts

a. In abstracto

26. The expression 'smart contract' origins from 1994. At that point in time, the established economic and communication systems did not allow for smart contracts to be exploited. Over time, mere speculation about potential opportunities advanced closer towards the actual realisation, since blockchain could provide a platform to underpin smart contracts.

The initial definition set forward by SZABO raised debate among authors.[54] Many discussions focus on issues regarding contesting terminology. BUTERIN, the creator of the most prominent smart contract platform Ethereum, defines smart contracts as: "systems which automatically move digital assets according to arbitrary pre-specified rules".[55] When compared to the original definition provided by SZABO, the emphasis is much more on the transfer of assets. From an overarching perspective, the literature distinguishes two categories of smart contract definitions.[56] The first category involves smart contracts as a specific technology (i.e. "smart contract code"). It refers to the usage of code, which is stored, validated and executed on the blockchain.[57] The second category of definitions adopt a more functional approach and refers to the actual application of the aforementioned technology. Smart contracts are defined as a complement of or substitute for legal contracts (i.e. "smart legal contracts").[58] The distinction between the notion smart contract code and the notion smart legal contract is of great relevance with regards to the legal and functional value of a smart contract.

i. Smart contract code – Code is law

27. Smart contract code relates to the notion of smart contracts as executable programs, thereby closely relating to the original definition of SZABO in 1994. The abilities of smart contracts within the notion of smart contract code are subject to the boundaries of the programming language in which they are written. The technical specifications of the underlying platform are important to assess smart

---

[54] *Supra*, 1; N. SZABO, "Smart Contracts", 1994, http://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/smart.contracts.html (Consulted on 18 November 2018).

[55] V. BUTERIN, "A next-generation smart contract and decentralized application platform", *supra* from 7, 2014, 1.

[56] M. ALHARBY e.a., "Blockchain-based smart contracts: A systematic mapping study", *supra* from 5, 2017, 127.

[57] J. STARK, "Making Sense of Blockchain Smart Contracts", *Coindesk*, 4 June 2016, https://www.coindesk.com/making-sense-smart-contracts (Consulted on 1 February 2019).

[58] *Ibid.*

contract capabilities. Smart contract code can be distinguished from other programs because smart contracts are recorded on and executed by a blockchain network.[59] This implies that smart contracts are connected to digital assets that reside on the blockchain network, such as cryptocurrencies or ownership titles, enabling transactions of these assets. The sole purpose of the word *contract* is to indicate a certain element of value embedded in these programs. Therefore, little attention is dedicated to the actual legal consequences of such programs, or more important, the question whether or not they are legal contract at all.

28. Defenders of the smart contract code definition no longer consider a law-enforcing government as a *conditio sine qua non* to regulate modern society. Already in 1993, LUHMAN drew a parallel between the true or false opposition specific to code on the one hand, and the opposition between legal and illegal specific to the law on the other hand.[60] To a larger extent, one could wonder whether the law itself is a fixed variable to take into account. Lawrence LESSIG was one of the most prominent authors on the theory of code-based regulation. He was responsible for the creation of the paradigm 'code is law': technology (i.e. code) can be used to enforce certain rules. LESSIG even suggested the possibility of codifying the law. This premise should, however, be nuanced. According to LESSIG, 'code is law' does not resemble to code equals law.[61] The paradigm is justified by the mutual capacity of law and code to regulate human behaviour through a series of instructions and rules. Code has the inherent capacity of self-execution: a specific intervention from the participants becomes redundant.[62] There's no longer a need to depend on the continued agency, loyalty or reliability of individuals.[63] This leads to the inevitable conclusion that for certain aspects of behaviour regulation, code can work more efficient than law.[64] Nonetheless, the other side of the coin should be taken into account as well: any decision made with regards to the development of such code will be affected by political and societal values of the programmer, rather than market values.[65]

---

[59] J. STARK, "Making Sense of Blockchain Smart Contracts", *supra* from 57, 4 June 2016, 4.
[60] N. LUHMANN, "Das recht der gesellschaft", *Frankfurt am Main*, 1993, 69-70, 165-213.
[61] L. LESSIG, *Code version 2.0*, New York, Basic Books, 2006, http://codev2.cc/download+remix/Lessig-Codev2.pdf, 342.
[62] *Ibid.,* 324.
[63] *Ibid.,* 342. The example of the launching of nuclear missiles is given by Lessig. Individual crew members disobeying the order to launch the missile would be subjected to a court-martial. Later on, the army preferred a new system that wired the missiles to a "single button" on the president's desk, increasing reliability.
[64] R.H. WEBER, "'Rose is a rose is a rose is a rose'–what about code and law?", *Computer Law & Security Review*, 2018, 703.
[65] *Ibid.,* 703.

ii. Smart legal contract – Law is code

29. The outlined notion of smart contracts as smart contract code received a fair amount of criticism by people active in financial and legal industries. This is mainly due to the misleading alleged correspondence of a *contract* with a *program.* Evidently, the former has legal consequences inherently attached to its existence. The same cannot be said about the latter, as code as such does not contain any legal value. However, there can be a certain interaction between the two concepts. Smart legal contracts require pieces of code, or smart contract code, to be implemented and to be executed if certain pre-defined conditions are met.[66]

30. Contract law is an ever changing area of law, developing in accordance with changing business models and technologies. Disruptive technologies such as the internet can highly affect the dynamics of contract law. However, the predominant definition of a contract throughout history has remained the same: "A contract is an agreement giving rise to obligations which are enforced or recognised by law. The factor which distinguishes contractual from other legal obligations is that they are based on the agreement of the contracting parties".[67] (emphasis added). Hence, conventional understanding has it that the law is the only source of the binding legal nature and enforceability of an agreement.[68] This view is diametrically opposed to the view of the defendants of the smart contract code definition, according to whom the role of the law can be (partially) assumed by code. In this regard, some legal scholars argue that smart contracts are a specific form of self-help: legally permissible conduct in the absence of an authority to prevent or remedy a civil wrong.[69] No recourse to a court is necessary to effectively execute the contract. This rather extreme point of view would deprive smart contracts of any possible legal capacity. A more moderate view acknowledges the possibility of legal capacity for smart contracts.[70] Scholars in favour of this position argue that smart contracts, although not enforced by law, can facilitate transactions of (digital) assets, therefore entering the realm of civil law.[71] Further elements, such as the precondition of the meeting of the minds of parties are used to underpin this theory.

---

[66] ISDA and LINKLATERS, "Smart Contracts and Distributed Ledger – A Legal Perspective", 2017, 5. https://www.isda.org/a/6EKDE/smart-contracts-and-distributed-ledger-a-legal- perspective.pdf (Consulted on 20 January 2019).
[67] G.H. TREITEL, *The law of contract*, Thomson/Sweet & Maxwell, 2003, 1.
[68] G. JACCARD, "Smart Contracts and the Role of Law", *Justletter IT*, 2017, 23, 9.
[69] A. SAVELYEV, "Contract law 2.0: 'Smart'contracts as the beginning of the end of classic contract law", *Information & Communications Technology Law*, 2017, vol. 2, 10.
[70] *Ibid.*; E. TJONG TJIN TAI, "Formalizing Contract Law for Smart Contracts", *supra* from 4, 2017, 3.
[71] A. SAVELYEV, "Contract law 2.0: 'Smart'contracts as the beginning of the end of classic contract law", *supra* from 69, 2017, vol. 2, 10.

31. In this regard, it should be noted that the correspondence between smart contracts and legal contracts implies the application of the general principles of the law of contract. It is still unclear how the current law of contract will adapt to regulate and enforce smart contracts. The applicability of doctrines such as establishing capacity, offer and acceptance, deceit and many more are considered to be some of the main legal challenges to be faced in the light of the commercial viability of smart contracts. The existing body of literature is aware of these challenges. They still constitute the main source of scepticism towards the viability of smart contracts.

iii. Legal framework

32. So far, only a certain amount of adaptations were made by legislators aiming to resolve these legal uncertainties. In the United States, a handful of States chose to implement smart contracts in their legislation. An example is the State of Arizona, which defined smart contracts as: "an event-driven *program*, with state, that runs on a distributed, decentralized, shared and replicated ledger and that can take custody over and instruct transfer of assets on that ledger" (emphasis added).[72] One can argue that this type of definition gravitates towards a smart contract code approach, denying the legal value of smart contracts.[73] In this regard, it is worth noting that the Illinois General Assembly is undertaking steps to introduce a Blockchain Technology act, which currently comprises a different type of definition of smart contracts.[74] The proposal defines smart contracts as: "a contract stored as an electronic record which is verified by the use of a blockchain", thereby recognizing the legality of smart contracts. Nonetheless, critics express a disapproval for such legal practises. It is claimed that the U.S. States are mainly incentivised to adopt such legislation to attract investors and entrepreneurs.[75] As a result, a legal patchwork is brought to existence, which increases chaos and legal uncertainty.[76] Countries need to implement uniform legislation in order to make these technologies work at their full potential.
The U.S. legislative initiatives share a certain policy choice. The proposed smart contract definitions include the existence of an underlying decentralised ledger. Apparently, legislators consider a decentralised ledger indispensable to constitute smart contracts.

---

[72] Article 5, 44- 7061, Chapter 26: Electronic Transactions of the Arizona Revised Statutes. https://www.azleg.gov/arsDetail/?title=44 (Consulted on 9 November 2018)
[73] *Supra,* 27.
[74] Section 5 of the Proposal of legislation HB5553 of the Illinois General Assembly of the 8th of January 2019, http://www.ilga.gov/legislation/BillStatus.asp?DocNum=5553&GAID=14&DocTypeID=HB&SessionID=91&GA=100, (Consulted on 24 March 2019).
[75] M. ORCUTT, "States that are passing laws to govern 'smart contracts' have no idea what they're doing", *MIT Technology Review*, 29 March 2018, https://www.technologyreview.com/s/610718/states-that-are-passing-laws-to-govern-smart-contracts-have-no-idea-what-theyre-doing/ (Consulted on 24 March 2019).
[76] *Ibid.*

iv. The necessity of a blockchain network

33. This leads to the question whether or not a blockchain network is necessary to classify a coded contract as a smart contract. The initial conception of the notion "smart contract" occurred before the term "blockchain" was coined. Moreover, even though most definitions assume that smart contracts and blockchain go hand in hand, not all authors agree to such assumption.[77] Using a decentralised ledger as a platform to embed smart contracts has several advantages. Primarily, the smart contract becomes immutable. Malicious attempts to alter the smart contract code would fail to do so.

34. The notion of computable contracts has been around for a longer time than BUTERIN's launch of Ethereum and the commodification of smart contract possibilities.[78] Especially in the financial industry, contracts have been structured as computer data records, rather than as written language statements.[79] One example is the Financial Information eXchange (hereafter: FIX). Initiated in 1992, FIX acts as an electronic communications protocol which facilitates real-time exchanges of information concerning security transactions and markets. Banks and other financial institutions rely on the protocol to trade electronically.[80] From 2001 on, research was devoted to the computability of legal obligations.[81]

35. One way to look at the relevance of a blockchain ledger is to analyse the notion 'smart contracts' as such. The most familiar part in the expression smart contracts is 'contracts'. According to the principles of European contract law, a contract is concluded when parties intent to enter into a binding legal relationship or bring about some other legal effect, and reach a sufficient agreement.[82] Important to note is that no specific contract form is required to confirm the legal existence of an agreement. Often referred to as the principle of *solo consensu,* parties are free to express their intentions to enter into an agreement in any form. This could be established orally, in writing, or in any other form. Hence, we argue that an agreement can be validly adopted in coded form as well. We can conclude that smart contracts are most definitely contracts from a legal perspective.

---

[77] M. RASKIN, "The Law and Legality of Smart Contracts", *supra* from 14, 2017, vol. 2, 306. ("Smart contracts are defined as agreements wherein execution is automated, usually by computers."); C.D. CLACK, V.A. BAKSHI en L. BRAINE, "Smart contract templates: foundations, design landscape and research directions", *Position paper*, 2016, https://arxiv.org/pdf/1608.00771v2.pdf. ("A smart contract is an agreement whose execution is both automatable and enforceable. Automatable by computer, although some parts may require human input and control. Enforceable by either legal enforcement of rights and obligations or tamper-proof execution").

[78] H. SURDEN, "Computable Contracts", *UCDL Rev.*, 2012. SURDEN mentions a "data-oriented contract" in which "the parties have expressed some part of their contractual arrangement as computer-processable data". Furthermore, contract are computable when "the parties have arranged for a computer to make automated, prima-facie assessments about compliance or performance".

[79] *Ibid.,* 640.

[80] For more, see https://www.fixtrading.org.

[81] W. MCGEVERAN, "Programmed Privacy Promises: P3P and Web Privacy Law", *NYU Law Review*, 2001, 1812.

[82] C. VON BAR e.a., *Principles, definitions and model rules of European private law: draft common frame of reference (DCFR)*, *supra* from 18, 2009, 108.

36. This leads us to the second part of the notion 'smart contracts', being the 'smart' aspect. This is a highly debated subject, as there's no consensus yet on the association of 'smart' in smart contracts with a blockchain based ledger. It might be sufficient to consider a contract 'smart' once its execution is automated, disregarding the platform upon which the contract is built. In this regard, we can argue that a contract becomes a smart contract by being able to execute itself or, in other words, make decisions without the need for human intervention.[83] Parties might prefer the adoption of smart contracts on a centralised ledger. For example, a supplier and a distributor who have been contracting for a long time might have developed a strong relationship of trust. They might consider optimising their agreements by using smart contracts through a centralised ledger.

Such perception allows a significantly larger concept of smart contracts, as it disregards the platform onto which smart contracts are built. We believe that the essential characteristics of blockchain, forming an immutable ledger on the one hand and rendering any form of trust redundant on the other hand, are not essential for the creation of smart contracts. On the contrary, they are mere modalities of smart contracts, increasing their functionality and security. As a consequence, smart contracts can be visually represented in the image below.[84]

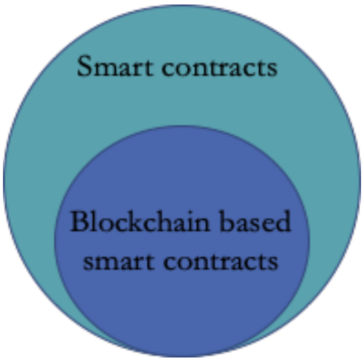


Figure 2 - Smart contract diagram. Blockchain based smart contracts comprise only a part of the entire cluster of smart contracts.

v. Synthesis

37. For the remainder of this research, the concept of smart contracts will be approached in a uniform matter. As a too restrictive approach with regards to the legality of smart contracts would deprive this research from its relevance and adequacy, it is assumed that smart contracts contain the inherent capacity of forming legally binding contracts. Such reasoning is based on the assumption that

[83] C.D. CLACK, "Smart Contract Templates: Legal semantics and code validation", *Journal of Digital Banking*, 2018, vol. 4, 339.

[84] Y. YU, *Contract as Code: Benefits and Challenges*, Ghent, University of Ghent, 2018, 11, https://lib.ugent.be/fulltxt/RUG01/002/479/450/RUG01-002479450_2018_0001_AC.pdf (Consulted on 1 May 2019).

parties act to create a legal relationship.[85] Moreover, contract laws do not explicitly prohibit expressing contract clauses in terms of code.[86] In the situation where parties agree on entering a smart insurance contract, they clearly show the intention to accept the legal consequences of such contract. As a consequence, the applied definition tends towards the smart legal contract notion.[87] The word *contract* implies more than a symbolic representation of value, as claimed by the defenders of the smart contract code interpretation.[88] On the other hand, the word *smart* is mere fiction: no contract is capable of determining its own clauses or replacing the human capacity to take rational decisions.[89] Therefore, the *smart* component to smart contracts represents its ultimate functionality: the capacity of self-execution. A final aspect to the synthesis definition in this research resides in the underlying platform upon which smart contracts rely. The integration of a distributed ledger technology provides several advantages which enable connectivity between parties that intrinsically lack trust towards each other. However, for a smart contract to be smart, a distributed ledger is not a precondition. Automated execution can be achieved by different means, such as centralised ledgers.

38. In summary, the definition of smart contracts used in this research is the following: *smart contracts are contracts which are characterised by the ability of self-execution through the use of code.*

39. Because of the new technological landscape that emerges with the rise of blockchain applications, the remainder of this research will approach smart contracts assuming that they are adopted on a blockchain based ledger. As *figure 2* indicates, blockchain based smart contracts are a part of a greater whole of smart contracts. We argue that, by discussing the former, we will cover the non-blockchain based smart contracts as well. The general principles that will be discussed in Title V also apply to smart contracts which are adopted on different digital ledgers.

b. In concreto

40. Disregarding any legal value smart contracts might have, they are essentially composed of a collection of instructions that perform a specific task when executed by a computer. These instructions are written in a programming language. Throughout computer science history, many programming languages have been developed. Each language has its own syntax and semantics, or vocabulary, containing sequences of words, numbers and punctuation that can form legible expressions.[90] Initially,

---

[85] For more, see M. GIANCASPRO, "Is a 'smart contract' really a smart idea? Insights from a legal perspective", *supra* from 7, 2017, vol. 6, 825-835; M. RASKIN, "The Law and Legality of Smart Contracts", *supra* from 14, 2017, vol. 2, 321-326.
[86] H. SURDEN, "Computable Contracts", *UCDL Rev.*, 2012, 656.
[87] *Supra*, 29.
[88] V. BUTERIN, "A next-generation smart contract and decentralized application platform", *supra* from 7, 2014, 1.
[89] B. SINGHAL e.a., "How blockchain works", *supra* from 29, 2018, 256.
[90] *Infra*, 117

source code is written by computer programmers as a human-readable form of code through the use of these expressions.[91] Subsequently, this source code is transformed into machine code by a compiler. The collection of instructions contained in the machine code, though not comprehensible for humans, can be executed directly by a computer. This subject is developed further in Title V.

41. Computer programmers mainly use the language Solidity to write smart contracts.[92] The language is heavily related to more common and established languages, such as Javascript, Java or C. The code is executed on a blockchain network which supports the programming language (i.e. Solidity on Ethereum). At present, most of the written smart contract code is deployed on the Ethereum network.[93] Solidity, being the source code of smart contracts, is compiled to a low level, stack-based machine code language, referred to as EVM code[94], which can be executed on the Ethereum network.[95] As an example of smart contract source code, following piece of code is provided, which establishes a decentralised auction for a certain item. Lower bids are returned to their bidder once a higher bid has been announced.[96]

[91] *Infra*, 115; M.J. ROCHKIND, *Advanced UNIX programming*, United States, Pearson Education, 2004, 1.1.
[92] Other popular languages are Serpent and LLL.
[93] C. DANNEN, *Introducing Ethereum and Solidity: Foundations of Cryptocurrency and Blockchain Programming for Beginners*, New York, Apress, 2017, 10.
[94] *Infra*, 115 ; EVM stands for Ethereum Virtual Machine. The EVM is the execution engine of Ethereum that works as the runtime environment for smart contracts. For more, see B. SINGHAL e.a., "How blockchain works", *supra* from 29, 2018, 257.
[95] B. SINGHAL e.a., "How blockchain works", *supra* from 29, 2018, 254.
[96] A.S. KOK, "Write A Simple Contract On Top Of Ethereum", *Coinmonks*, 2 May 2018, https://medium.com/coinmonks/write-a-simple-contract-on-top-of-ethereum-92b543594e84 (Consulted on 21 February 2019).

```
pragma solidity ^0.4.19;
contract Auction {
  address public manager;
  address public seller;
  uint public latestBid;
  address public latestBidder;

  constructor() public {
    manager = msg.sender;
  }

  function auction(uint bid) public {
    latestBid = bid * 1 ether;
//1000000000000000000;
    seller = msg.sender;
  }

  function bid() public payable {
    require(msg.value > latestBid);

    if (latestBidder != 0x0) {
      latestBidder.transfer(latestBid);
    }
    latestBidder = msg.sender;
    latestBid = msg.value;
  }

  function finishAuction() restricted public {
    seller.transfer(address(this).balance);
  }

  modifier restricted() {
    require(msg.sender == manager);
    _;
  }
}
```

Such example illustrates that, even though certain words seem to be familiar, the interpretation of a smart contract is rather difficult for a non-qualified person. We will propose a solution for such interpretation difficulties in Title IV.[97]

2.2. Beyond Bitcoin

42. As mentioned before, Bitcoin is considered the first generation of blockchain applications. This paragraph will describe the evolution from the first generation to the second generation (i.e. smart contracts).

Bitcoin pushed blockchain in the technological arena and showcased its potential. However, the underlying protocol of Bitcoin facilitates rather straightforward transactions and lacks the capacity to provide more elaborate blockchain transaction applications. The scripting language embedded in the Bitcoin platform has significant limitations with regard to higher levels of functionality.[98] Moreover, the blockchain network supporting Bitcoin transactions only comprises data on the most recent asset holder and subsequently, occurring Bitcoin transfers. Therefore, Bitcoin transactions can be considered

[97] *Infra,* 71.

[98] *Supra,* V. BUTERIN, "A next-generation smart contract and decentralized application platform", *supra* from 7, 2014, 12.

as downsized prototype versions of smart contracts. A few steps were taken to enable the creation of more complex smart contracts on a blockchain network.

a. Turing completeness

43. As mentioned before, the pioneering Bitcoin blockchain provided the opportunity to facilitate straightforward transactions. The original Bitcoin protocol was written by NAKAMOTO in C++. Although this programming language is considered to be rather archaic and inflexible, it contained sufficient technical advantages for NAKAMOTO to prefer it to other, more modern languages. One of the consequences of using C++ was the inability of the Bitcoin protocol to solve every computational problem. This is mainly caused by C++'s lack of capacity to execute loops (i.e. the capacity of executing pieces of code over and over again).[99] The original creators of the Bitcoin script intended to prevent an overload of the blockchain network, that could occur due to code executing itself millions of times in a row. As a result, the Bitcoin scripting language could only execute the same code several times in a row, if that particular code was copied. Hence, the Bitcoin scripting language was not Turing complete: only specific types of computational problems could be solved. This imposed stringent limitations to the potential applications of smart contracts, which required a higher degree of complexity of programming language.

44. These problems were resolved with the creation of blockchain networks programmed with a more sophisticated and flexible language than C++. 2015 marked the birth of the Ethereum blockchain, announcing the next major evolution in blockchain-based applications. Its founder Vitalik BUTERIN created a blockchain network with a built-in Turing-complete programming language, allowing developers to create decentralised arbitrary consensus-based applications. To overcome the issue of an overload of the blockchain network, the developers of Ethereum introduced fees for each operation. In order to execute the smart contract, an amount due must be paid by the creator. The amount of computational power required to run the program is expressed in the unit Gas. Each line in the programming language requires a certain amount of Gas upon executing. The amount paid by the creator serves as an incentivizing reward for the miner solving the proof-of-work and, as a consequence, adding another block to the blockchain.[100]

b. Binary state

45. Advocates of consensus-based decentralised applications were confronted with several other problems in the Bitcoin blockchain. One of the main issues apart from the blockchain being Turing incomplete was the lack of state of Bitcoins. Bitcoins can either be spent or unspent; there is no other

---

[99] *Ibid.* 12
[100] *Supra*, 18

alternative multi-level state beyond that. This binary state rendered multi-stage option contracts, computational bounties or decentralised exchange offers impossible. Ethereum and the following next generation blockchain networks resolved this issue through a script supporting a multi-level state.

c. A next generation blockchain: new applications

46. Three categories of applications were now made possible.[101] Firstly, as the Ethereum blockchain network created its own cryptocurrency Ether, financial transactions similar to Bitcoin transactions could exist. Moreover, users were offered a whole new range of possibilities to manage their money. As an extension to the existing Bitcoin blockchain, Ethereum opened the door to the second category of transactions: semi-financial applications with a heavy non-monetary side. An example would be the licensing of intellectual property rights through an Ethereum application which determines the royalty fee and the license specifications.[102] Lastly, non-financial applications might be created as well. Specifically designed voting mechanisms could benefit from a decentralised approach where transaction costs are high.[103] A heavily debated illustration of the third category are so-called decentralised autonomous organisations, or DAO's, which will be discussed later.

2.3. Oracles

47. As smart contracts reside in a blockchain network, their existence is confronted with the opportunities that result from such digital online environment. Moreover, an interplay with other smart contracts or data services can affect the conditional execution of smart contracts.[104] In other words, smart contracts can intentionally rely on other data, which is gathered after the creation of the smart contract itself, as a precondition upon execution. In this regard, it is crucial to determine and rely on data that is accurate and true. The opposite might compromise the intention of the contracting parties and the desirability of the result after the execution of the contract. The providers of such accurate data are called *oracles*.[105] Oracles are the third-party agents that bridge the gap between the decentralised digital world of smart contracts and the real world. Oracles can be both machine-based and human-based. They find and verify data, such as plane delay times, weather conditions, results of sports events,

[101] V. BUTERIN, "A next-generation smart contract and decentralized application platform", *supra* from 7, 2014, 19.
[102] The platform Ujo Music, a pioneer Ethereum blockchain distribution service, is a good illustration of such application. Its purpose is to create a transparent and direct connection between artists and listeners. Song details are recorded in a blockchain based database. Through smart contracts, these songs can be purchased and downloaded. The compensation streams make it to the rightful owners, however differentiating separate artist contributions to the song. For more, see https://ujomusic.com.
[103] For more, see C. VAN DER ELST en A. LAFARRE, "Bringing the AGM to the 21st Century: Blockchain and Smart Contracting Tech for Shareholder Involvement", *European Corporate Governance Institute (ECGI) - Law Working Paper No. 412/2018*, 2018, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3219146.
[104] V. GATTESCHI, F. LAMBERTI, C. DEMARTINI, C. PRANTEDA en V. SANTAMARÍA, "Blockchain and smart contracts for insurance: Is the technology mature enough?", *Future Internet*, 2018, vol. 2, 20, 4.
[105] ISDA e.a., "Smart Contracts and Distributed Ledger – A Legal Perspective", *supra* from 66, 2017, 18.

elections etc.[106, 107] For machine-based oracles, software oracles can be distinguished from hardware oracles. The former extracts data from information available online, such as prices of commodities. The latter requires information directly from the physical world, such as data from supply chain sensors, registering the temperature of goods during their transportation.[108]

48. Recent developments might accelerate the possibilities to include oracle data in smart contracts. Over the past few years, a lot of attention has been devoted to the *Internet of Things,* or IoT. This notion is used as a catch-all term for the increasing interconnectivity of people and devices in a human-made virtual environment.[109] Smart devices, sensors, human beings and any other object that is aware of its context and is able to communicate with other entities will be considered to reside in the Internet of Things ecosystem.[110] The combination of smart contracts with the Internet of Things sector is promising, paving the way for new business models and distributed applications.[111] This can be illustrated by an example in the energy sector. On certain occasions, the renewable energy produced by solar panels might be redundant for the owner to meet his own energy needs. Therefore, it could be commercially viable to sell this energy surplus to requesting neighbouring parties. LO3 Energy, a New York-based start-up, provides for a blockchain based self-executing solution. The output surplus of the registered solar panels is recorded and transmitted to a blockchain network, relying on the IoT ecosystem. Consequently, it is sold via smart contracts to parties in demand for energy.[112]

49. Apart from the distinction between human-based oracles and machine-based oracles (i.e. hardware and software oracles), several stages of oracles can be observed, dependent on their complexity. The first category is a trusted user. Such user refers to an internal piece of code, running on a blockchain ledger that verifies whether or not an event has taken place. The trusted user then feeds data into a smart contract which interacts and, subsequently, executes a transaction based on the results of the data. Hence, a trusted user can be described as a smart contract version of an arbitrator or a third-party certification.[113] A more complex version of an oracle is a trusted data feed.[114] In this case, online, but off-blockchain data sources are consulted. Example of such sources are global air traffic

---

[106] J. DOURLENS, "Oracles: bringing data to the blockchain", *Ethereum Developers*, 9 October 2017, https://ethereumdev.io/oracles-getting-data-inside-blockchain/ (Consulted on 19 February 2019).
[107] For more, see http://www.oraclize.it
[108] J. DOURLENS, "Oracles: bringing data to the blockchain", *supra* from 106, 9 October 2017.
[109] R. BUYYA, A.V. DASTJERDI en F. KHODADADI, "Internet of Things: An Overview" in *Internet of Things: Principles and paradigms*, Australia, Elsevier, 2016, 3.
[110] *Ibid.*
[111] K. CHRISTIDIS en M. DEVETSIKIOTIS, "Blockchains and smart contracts for the internet of things", *Ieee Access*, 2016, 2292, ieeexplore.ieee.org/document/7467408/ (Consulted on 21 February 2019).
[112] For more, see https://lo3energy.com
[113] J. GRIMMELMAN, "All Smart Contracts Are Ambiguous", *Penn Journal of Law and Innovation*, 2019, 14, https://ssrn.com/abstract=3315703 (Consulted on 17 April 2019).
[114] *Ibid.*

databases.[115] The third and most complicated category of oracles is a consensus oracle. A group of users will act as an oracle, based on an achieved consensus between the members of the group.[116]
It should be noted that, while oracles enable real world interconnectivity and therefore increase smart contract opportunities, they also might constitute a smart contract's Achilles' heel. Several concerns are raised with regard to the reliability of these oracles. As mentioned before, blockchain is lauded for its ability to eliminate the need for trust. However, as these oracles feed data into the blockchain, they might compromise the reliability of the ledger. This happens when the oracle itself is corrupted. With regard to the third category, the reliability of the oracle is only as solid as the thoroughness of the underlying consensus protocol.[117]

CHAPTER 3. PRELIMINARY CONCLUSION

50. This Title attempted to provide an overview of the technical and legal fundamentals upon which smart contracts are built. In doing so, several research questions were answered. Firstly, the function and benefits of blockchain were clarified. Providing an immutable and trustworthy ledger for transactions, blockchain-based ledgers might endow the missing link to construct secure smart contracts.
Subsequently, several definitions of smart contracts were described. We can conclude that there is a remaining discord between authors. For a large part, this is caused by the opposition between defendants of the smart contract code definition and the defendants of the smart legal contract definition. In our synthesis definition, smart contracts are granted the capacity of forming legally binding contracts. In addition, an underlying decentralised ledger is not considered to be a precondition for the creation of smart contracts. Instead, it was made clear that the role of blockchain based ledgers resides in facilitating more secure and more functional smart contracts. Lastly, the connection between the digital world and the physical world through the use of oracles was described. Being both a blessing and a curse for blockchains, oracles are rather delicate and should be used with caution.
Title IV attempted to provide insights in the technical and legal background of smart contracts. In doing so, the foundation was laid to transform contracts into smart contracts.

[115] For an example, see *infra,* 89.
[116] J. GRIMMELMAN, "All Smart Contracts Are Ambiguous", *supra* from 113, 2019, 14.
[117] *Ibid.,* 15.

TITLE IV. A PARADIGM SHIFT: NEXT GENERATION CONTRACTS

51. The narrative of contracts is characterised by a remarkable evolution throughout history. Several milestones can be distinguished in this development: from oral contracts to written contracts to digital contracts. However, the path of evolution of contracts has been heading into a different direction when compared to technological developments. Intel co-founder Gordon Moore predicted in the 1980's that the number of transistors in a dense integrated circuit would be doubled about every two years. This has a direct impact on the amount of computational power, and therefore, the complexity of the problems which can be solved by computer processors. His prediction has stood the test of time, although the pace of innovation has slowed down over the last years.[118]

Whereas technology advances in an innovative and relentless way and more and more complex problems can be solved, the general format and structure of contracts has remained relatively the same for the last thirty years.[119] Contracts were, and still are, complex, impenetrable, highly contentious and time-consuming instruments.[120] Often they result in linguistically inaccessible documents; written by lawyers, for lawyers.[121] This reflects a unilateral perspective on the formation and execution of contracts: lawyers are rather reluctant to tolerate any input coming from other disciplines, such as computer science. Lawyers assume they have to anticipate every possible source of conflict. Simplicity is considered a risk, as it may imply ambiguity which might lead to conflicts and litigation.[122]

52. To answer the central research question of this dissertation, a deeper understanding should be developed on how contracts are drafted on the one hand and which tools are available to support such drafting on the other hand. With such knowledge, we can assess how an integration of smart contracts in the current contractual landscape might occur.

Recently, the aforementioned paradox between the dynamics of technology and the rigidity of contracting practices have been challenged by several initiatives. This Title will provide insight in the most prominent proposals to create more efficient, transparent and effective contracts and contract

[118] C.A. MACK, "Fifty years of Moore's law", *IEEE Transactions on semiconductor manufacturing*, 2011, vol. 2, 202–207.

[119] R. UNSWORTH, "Smart Contract This! An Assessment of the Contractual Landscape and the Herculean Challenges it Currently Presents for 'Self-executing' Contracts" in *Legal Tech, Smart Contracts and Blockchain*, Singapore, Springer, 2019, 18.

[120] *Ibid.*

[121] T. BARTON, H. HAAPIO, S. PASSERA en J. HAZARD, "Successful Contracts: Integrating Design and Technology" in *Legal Tech, Smart Contracts and Blockchain*, Singapore, Springer, 2019, 65.

[122] *Ibid.,* 87

practices. Chapter one will describe the interaction between the traditional 21st century contract model and technology. Such traditional contract model refers to the contracts as we know them: written in natural language.

Moreover, chapter two will attempt to explore opportunities to create contracts which are tailored to both humans (i.e. not only lawyers) and machines. Subsequently, we will present a new hybrid contract model, by integrating smart contracts and other technological opportunities into the existing contracting practices. This 'hybrid' contract model aims to form the foundation for next generation contracts (i.e. contracts for humans and machines). Such contract model will be used in the remainder of this research as a benchmark for smart contract implementation.

Finally, chapter three will delineate potential use cases for smart contracts. Furthermore, an overview is given of the remaining obstacles that prevent smart contracts from delivering their full potential.

CHAPTER 1. 21ST CENTURY CONTRACTING PRACTICES AND TECHNOLOGY

53. This chapter will describe how technology has been supporting and innovating the established commercial contracting practices of the 21st century. Contracting practices can be described as the processes which are maintained by contracting parties through the creation, the management and the execution of contracts. Such contracting practices will be evaluated from a technological perspective. An analysis of the evolutions in legal drafting falls beyond the scope of this research. Both contract management software and artificial intelligence integration have been used to enhance the efficiency of contracting practices of businesses.

1.1. Contract management software

54. Contract management refers to the perspective which analyses and coordinates an organisation's contracting process, from the initial negotiations to the termination of the contract. It aims to:

- reduce fragmentation of contracts within the organisation;
- improve contract quality, legal compliance, consistency and efficiency;
- facilitate a better collection of information with regard to strategic decisions.[123]

Contracts are a core asset of every organisation. An adequate contract management can play a significant role in terms of value creation and efficiency. The general discipline of contract management has been around for some years now. A substantial amount of organisations still do not

[123] C.E.C. PARIS, "Contract Management: Design Parameters and Challenges to Implementation", *Scandinavian Studies in Law*, 2010, 196, http://www.scandinavianlaw.se/pdf/49-10.pdf (Consulted on 27 February 2019).

realize the potential benefits related to the implementation of contract management systems. This leads to "value leakage": the value of a contract decreases between 17 and 40 percent of its total value.[124] Most organisations don't have a comprehensive database containing the information which is adopted in contracts. This increases employment costs, as it requires more manpower to examine contracts individually. For example, fixed-term contracts which the organisation would like to renew should be monitored with respect to their termination date. If this organisation is a large technology company which entered into a vast amount of complex outsourcing contracts, time expenditure and costs will raise substantially. Moreover, unsystematic contracting practices lead to an increased risk on litigation. An organisation does not have an effective risk management when it doesn't control and understand the obligations which are imposed by the contracts it enters into.[125]

55. With the rise of modern IT-infrastructures, contract management can be facilitated in a more convenient way. Software applications such as Agiloft and Concord provide a cloud-based platform to structure contracts, gain actionable insights and automatically integrate contract information into related business processes.[126] This leads to an increased consistency in contracts. Moreover, in the process of integrating technology into contracting practices, efficient contract management software can play the role of catalyst towards further optimisation.

56. Contract management can be perceived as a first step towards the integration of smart contracts. Parties who are conscious about their contracts and the resulting rights and obligations might have a substantial advantage in implementing new technologies in their contracting practices.

1.2. Artificial intelligence integration

57. Similar to blockchain and big data, artificial intelligence, or AI, has been a buzzword that appeared in all kinds of articles and contexts. Without any prejudice to its potential, one should be cautious about using the term artificial intelligence as a substitute for other concepts such as automated machines. Artificial intelligence can be defined as: "the design and setting of machines that mimic (also but not only) cognitive functions that humans associate with their own intelligence, such as learning and reasoning, planning and problem solving".[127] The term itself was coined in 1956 during the

---

[124] P. LIDDELL, M. SMART, C. RICHARDS en A. GIBBY, "Supply chain capacity management – the key to value", *KPMG Australia*, 21 March 2017, https://home.kpmg/au/en/home/insights/2017/03/supply-chain-capacity-management.html (Consulted on 27 February 2019); K. POTTS, "Driving better fiscal management and revenue recognition with enterprise contract management technologies", *IACCM*, 15 December 2008, https://www.iaccm.com/resources/?id=8026 (Consulted on 27 February 2019).
[125] C.E.C. PARIS, "Contract Management: Design Parameters and Challenges to Implementation", *supra* from 123, 2010, 196.
[126] For more, see https://www.agiloft.com; https://www.concordnow.com.
[127] U. PAGALLO, M. CORRALES, M. FENWICK en N. FORGÓ, "The Rise of Robotics & AI: Technological Advances & Normative Dilemmas" in M. CORRALES, M. FENWICK en N. FORGÓ (eds.), *Robotics, AI and the Future of Law*, Singapore, Springer, 2018, 5, https://doi.org/10.1007/978-981-13-2874-9_1.

Dartmouth Conference by John MCCARTHY. It was, however, only after the first decade of the twenty-first century that the concept gained more attention, due to the capacity of saving and interpreting more data and the use of advanced algorithms. As an umbrella term, artificial intelligence includes multiple aspects such as machine learning, deep learning, natural language processing and many more. By using and combining these techniques, artificial intelligence enables computers not only to fetch and display the data, but to make decisions based on the data as well. An example would be the spam filter of an e-mail inbox. Through natural language processing and machine learning, the filter recognizes specific words and combinations of sentences in order to classify mails as spam.

58. Just like the spam filter in an e-mail environment, the same application could be used to extract and classify contract clauses. Whereas traditional contract management software is used to manually determine contract renewal dates and negotiation terms, artificial intelligent systems manage to execute the same tasks automatically. Contract types such as non-disclosure agreements or outsourcing agreements can be distinguished based on pattern recognition of how the document is drafted.[128] The same type of contract clauses can be compared, encouraging more consistency in the process of drafting new contracts on the one hand and an improved tool to enhance future negotiation positions on the other hand.[129] Discrepancies between best practice clauses and contract proposals by other parties are highlighted. Moreover, AI can locate potential risks in these clauses, replacing more inefficient risk assessment teams.[130] Software facilitating the use of these technologies is already available on the market. An example is Kira, which applies machine learning software that identifies, extracts and analyses text in contracts and other documents.[131] Some of the world's largest professional service firms already have adopted the software, illustrating its promising opportunities.[132]

59. We will elaborate on the integration of AI technologies, such as machine learning and natural language processing throughout the remainder of this research.

---

[128] B. RICH, "How AI Is Changing Contracts", *Harvard Business Review*, 12 February 2018, https://hbr.org/2018/02/how-ai-is-changing-contracts (Consulted on 27 February 2019).

[129] M.-M. BUES en E. MATTHAEI, "LegalTech on the Rise: Technology Changes Legal Work Behaviours, But Does Not Replace Its Profession" in *Liquid legal: Transforming Legal into a Business Savvy, Information Enabled and Performance Driven Industry*, Singapore, Springer, 2017, 103.

[130] B. RICH, "How AI Is Changing Contracts", *supra* from 128, 12 February 2018.

[131] For more, see https://kirasystems.com (Consulted on 2 March 2019).

[132] ARTIFICIAL LAWYER, "Allen & Overy Formally Adopts Kira Systems for AI Doc Review", *Artificial Lawyer*, 10 July 2018, https://www.artificiallawyer.com/2018/07/10/allen-overy-formally-adopts-kira-systems-for-ai-doc-review/ (Consulted on 2 March 2019); M. TOM, "The robots keep coming: DLA Piper makes major AI play with Kira software deal", *Legal Business*, 14 June 2016, https://www.legalbusiness.co.uk/blogs/the-robots-keep-coming-dla-piper-makes-major-ai-play-with-kira-software-deal/ (Consulted on 2 March 2019).

CHAPTER 2. A NEW CONTRACT MODEL FOR HUMAN AND MACHINE

60. In the previous chapter, two forms of technology were discussed that have been increasing the efficiency of established contracting practices. However, these traditional contracting practices still rely on the assumption of contracts in a written form. Such traditional contracts are usually difficult to understand or to draft for people without any legal background. This chapter will explore the opportunities towards a different perception on the nature of contracts. The paradigm of written contracts composed in difficult 'legalese' might be directed in two new directions: towards both intelligible user-centred contracts and machine-centred contracts. Contracts could be understandable for both people and machines through the use of contract design on the one hand and code on the other hand, thereby resulting in a genuine paradigm shift.

This chapter then will integrate these elements into a new contract model, tailored for both human and machine. The third part of this research, or Title V, will use the model to assess whether or not contract clauses can be transformed to smart contract clauses. Depending on the outcome of such assessment, the contract clauses will be integrated in the model.

Finally, this chapter will give an overview of potential use cases for smart contracts. We will provide a number of case studies that demonstrate the opportunities.

2.1. Desirability of a paradigm shift

61. Even though the efficiency of contracts has been affected positively by aforementioned technological developments (i.e. contract management software and AI integration), there is still room for further improvement. A study of the IACCM, or the International Association for Contract & Commercial Management, indicates ten pitfalls of current contracting and commercial practice causing an average value erosion of contracts of 9.2%.[133] Among these pitfalls, the limited use of technology and the ambiguity of modern contracts are mentioned. This chapter will deal with both of these aspects. Evidently, smart contracts or any other technological paradigm shift will not be able to provide an exhaustive answer to every issue related to the value erosion of contracts. However, any decrease of this percentage could reduce costs substantially. Businesses are tempted to overestimate the intrinsic value of some technologies, due to their alleged all-inclusive high resolving character and the hype which surrounds them. Nonetheless, only if the costs are significantly lower than the benefits which an

[133] T. CUMMINS, "Commercial agility and creativity through contract simplification", Presentation at first international conference on contract simplification, Rüschlikon, 29 March 2016, slide 8, http://media.swissre.com/documents/Presentation_Tim_Cummins.pdf (Consulted on 4 March 2019).

adoption of a technology could bring, smart contract implementations are opportune.[134] A specific economic analysis of the cost-effectiveness of smart contracts goes beyond the scope of this research. Still, based on the scale of the industries in which smart contracts could play a role on the one hand, and on the general acceptance among (academic) writers of their unique cost-saving features on the other hand, it is assumed that smart contracts are desirable to a certain extent.[135]

2.2. Legal design – *User-centred contracts*

62. In recent years, a countermovement against the increasing complexity and illegibility of contracts has been established. This should be seen in the light of a broader context, which was given the title "legal design": an umbrella term for merging forward-looking legal thinking with design thinking.[136] BRUNSCHWIG was one of the first authors to raise the question whether contracts should necessarily be written.[137] Advocating a paradigm shift, certain authors propose a more visualised and simplified version of contracting.[138] The underlying reasoning is the intended independence from experts (i.e. lawyers) to translate needs of contracting parties, and the enhancement of overall attractiveness of contracts.[139] Through the creation of a user-friendly interface, contracting parties would be able to adjust certain parameters and insert the desired contract specifications.[140] Furthermore, an artificial veil could be raised to distinguish the user-friendly front side from the more complex legal back side. HAGAN refers to this practice as: "Be Simple on the Front, and Smart at the Back".[141] Hence, contracts would be split up into different modules, each directed towards a specific category of individuals. The front side, or the user-oriented side, can be displayed as a clear-cut version of a

---

[134] C. MULLIGAN e.a., "Blockchain beyond the hype; a practical framework for business leaders", *supra* from 13, 2018.
[135] E.g. the healthcare industry, the IoT data industry and the supply chain industry. For more, see K.N. GRIGGS, O. OSSIPOVA, C.P. KOHLIOS, A.N. BACCARINI, E.A. HOWSON en T. HAYAJNEH, "Healthcare Blockchain System Using Smart Contracts for Secure Automated Remote Patient Monitoring", *Journal of medical systems*, 2018, vol. 7, 130, https://www.ncbi.nlm.nih.gov/pubmed/29876661 (Consulted on 4 March 2019); A. SULIMAN, Z. HUSAIN, M. ABOUOUF, M. ALBLOOSHI en K. SALAH, "Monetization of IoT data using smart contracts", *IET Networks*, 2018, https://www.researchgate.net/publication/326718966_Monetization_of_IoT_Data_using_Smart_Contracts (Consulted on 4 March 2019); P. LIDDELL e.a., "Supply chain capacity management – the key to value", *supra* from 124, 21 March 2017.
[136] M. CORRALES, M. FENWICK en H. HAAPIO, "Digital Technologies, Legal Design and the Future of the Legal Profession" in *Legal Tech, Smart Contracts and Blockchain*, Singapore, Springer, 2019, 6.
[137] C. BRUNSCHWIG, "Multisensory law and legal informatics: a comparison of how these legal disciplines relate to visual law", *Jusletter IT*, 2011, http://jusletter-it.weblaw.ch/en/issues/2011/104/article_324.html (Consulted on 4 March 2019).
[138] G. BERGER-WALLISER, T.D. BARTON en H. HAAPIO, "From visualization to legal design: A collaborative and creative process", *American Business Law Journal*, 2017, vol. 2, 347–392.
[139] T. BARTON e.a., "Successful Contracts: Integrating Design and Technology", *supra* from 121, 2019, 66.
[140] *Ibid.*
[141] M. HAGAN, "6 Core Principles for Good Legal Design", *Medium*, 7 November 2016, https://medium.com/legal-design-and-innovation/6-core-principles-for-good-legal-design-1cde6aba866 (Consulted on 4 March 2019).

contract, accompanied by visual contract summaries and visual dashboards.[142] Additionally, the actual output on the front side can be subdivided in accordance with the relevant information that each user needs to incorporate the contract into his business and to execute the clauses. The back side represents the more complex legal framework. Empirical studies indicate that such a distinction contributes to an enhanced comprehension and, consequently, to better engagement and contract compliance.[143] Also, from the perspective of a lawyer, the efforts which are necessary to draft the contract will be reduced, because the client's awareness and conception of the content of the contract will be improved. Clients will develop a deeper understanding of the key points and pitfalls of contracts by repeatedly interpreting contract design-based overviews. Therefore, more precise instructions can be provided by clients with respect to their intentions and expectations. An enhanced collaborative relationship can be established between a lawyer and his client. Such practices might steer present-day contracting practices away from the adage "contracts written by lawyers, for lawyers".

2.3. Smarter contracts – *Machine-centred contracts*

a. Nick Szabo's smart contracts

63. In line with the design-related developments as described above, a second paradigm shift is facing the break of dawn. Whereas the introduction of legal design still adheres to human legible contracts, advocates of smart contracts go beyond this cornerstone by launching the idea of machine-centred contracts. Simplification of contracting practices is for them not an end in itself, as the focus shifts more towards efficiency in the execution phase of the contract. Going beyond a written contract and an online contract, smart contracts are deprived from any form of natural language what so ever. Human intervention is limited to the negotiation and the conclusion of the contract. From that point on, the machine will take over and execute the code adopted in the contract. Evidently, automation of contract processes has already been integrated in a variety of business operations (e.g. shares are traded through automated systems). However, such automation is limited to certain operations of the contract.

64. Smart contracts should be distinguished from E-contracts, or online contracts. The latter does not have to ability of self-execution. In the example of an online purchase of a pair of shoes, the buyer enters into an online contract with the distributor. General conditions are integrated in the internet contract, and are usually accepted by the buyer by checking a "I have read and agree to the Terms of Use" box. However, if the distributor does not perform his obligation of delivering the shoes, thereby violating the Terms of Use, the buyer will have to take the initiative to file a complaint or claim a refund.

---

[142] T. BARTON e.a., "Successful Contracts: Integrating Design and Technology", *supra* from 121, 2019, 67.
[143] S. PASSERA, "Beyond the wall of text: How information design can make contracts user-friendly" in *Design, user experience, and usability: users and interactions. Lecture Notes in Computer Science*, Cham, Springer, 2015, 341–352.

To a certain extent, there is still the possibility of breach of contract if the seller does not comply with his obligations or any remedy mechanisms. On the contrary, smart contracts can mitigate these risks by inserting predetermined remedy mechanisms. If the distributor does not comply with his obligation, due to insolvency, fraud or negligence, the smart contract can determine an alternative supplier to re-order the same pair of shoes and consequently transfer the value accordingly. Also, if the buyer has already paid the price for the shoes, reimbursement can be executed automatically.

b. Smarter contracts

65. Nick SZABO proposed a rather radical idea of a self-executing contract: the contract as a whole is considered to be an entirely automated process. The sole authentic version of the contract resides on a blockchain network as a piece of computer program. However, some authors suggest more pragmatic solutions as an alternative to this purist notion of smart contracts.[144] They argue that not everything can or should be automated, as the self-execution of certain clauses might not be efficient or desirable in terms of anticipated risks.

In the example of self-driving cars, it might be more appropriate to limit human intervention to specific key moments (e.g. difficult traffic situations) as opposed to exclude human control entirely, as this would reduce production costs and lower the potential risk for self-driving cars to cause an accident. Similarly, for more complex contractual relationships, it will often not work to redesign the relationship in an exclusively automated form, without any kind of human built-in recourse mechanism.

This results in a hybrid model, or so-called "smarter contracts", in which human intervention remains relevant and in which there is still recourse to legal language, but only to a certain extent.

Two types of such hybrid smarter contracts model are distinguished in the white paper by ISDA and Linklaters: an external type and an internal type.[145]

i. External smart contract

66. The external smart contract model refers to a dualistic contractual construction in which certain clauses are extracted from the traditional contract and, subsequently, transformed into code which is then embedded in a blockchain network. The purpose of this setup is to automate certain conditional logic elements of the initial contract so the required actions execute automatically upon fulfilling the relevant conditions.[146] The traditional contract is superior to the coded extract, which forms an external part of the contract, thereby relating to the smart contract code definition.[147] The code merely represents an automated tool for execution, as opposed to a legal binding instrument as such. In the event ambiguity

[144] ISDA e.a., "Smart Contracts and Distributed Ledger – A Legal Perspective", *supra* from 66, 2017.
[145] *Ibid.*
[146] *Ibid*, 14.
[147] *Supra,* 27.

arises, the written contract would take precedence.[148] For instance, in case of an insurance contract, the contract can externalise certain clauses which are embedded as code on a blockchain network. Consequently, these clauses could automatically trigger reimbursements (e.g. based on weather reports or flight delay details) leading to an acceleration of the processing of claims and a reduction of operating costs for insurance companies.[149] However, fraudulent practices by the insured will still be governed by the traditional contract. The general conditions of such contract can provide remedies to exercise recourse against malicious behaviour.

67. As mentioned before, a certain amount of automation has already been integrated in contractual processes. Due to the mere instrumentalist non-legal conception of code in the external model, it seems to be only a small step further then the mechanisms that were in place before.[150] It is therefore doubtful whether the introduction of external smart contract models will heavily impact lawyers' practices. Moreover, it is questionable whether setting up or using an existing blockchain network will generate an actual economic advantage, as the selling point (i.e. redundancy of legal enforcement) might be compromised by legal actions reversing automated execution. Another issue that may rise is related to the subordination of the smart contract component to the natural language contract. Any renegotiated adjustments to the latter imply alterations of the former as well. To safeguard the authenticity of the contract, such modifications of the smart contract component should be realised promptly. Hence, there also must be a way to change the blockchain if required.[151]

ii. Internal smart contract

68. Contrary to the external smart contract model, in an internal smart contract model the legal contract as a whole also encompasses any coded sections of the contract. Code is given the same legal value as natural language. Evidently, this has significant consequences for the legibility of the contract because of the creation of a patchwork of legal provisions in either code or natural language. As an alternative, the natural language contract could refer to a piece of code which is not directly adopted in the contract, and announce its legal effect between parties.[152]

69. We believe that an internal smart contract model is the ideal solution to combine both the benefits of auto-execution on the one hand and the ability to maintain a natural language contract on

---

[148] *Ibid.*
[149] V. GATTESCHI e.a., "Blockchain and smart contracts for insurance: Is the technology mature enough?", *supra* from 104, 2018, vol. 2, 12.
[150] ISDA e.a., "Smart Contracts and Distributed Ledger – A Legal Perspective", *supra* from 66, 2017, 13.
[151] M. DEMEYER, *Blockchain Technology And Smart Contracts From A Financial Law Perspective*, Ghent, University of Ghent, 2018, https://lib.ugent.be/fulltxt/RUG01/002/479/332/RUG01-002479332_2018_0001_AC.pdf (Consulted on 13 March 2019).
[152] ISDA e.a., "Smart Contracts and Distributed Ledger – A Legal Perspective", *supra* from 66, 2017, 13.

the other hand. Parties cannot attempt to reverse the agreed upon coded provisions by contesting the interpretation of the agreement, based on a preceding natural language contract. The coded clauses are an authentic part of the contract. An external smart contract approach would contravene our proposed definition of smart contracts, as smart contracts would be deprived from any legal value (i.e. smart contracts would be no longer contracts).

70. The adoption of an internal smart contract model leads to an interpretation disadvantage. The contract as a whole will form a patchwork of both natural language clauses and coded clauses. To resolve such problem of legibility, we propose a Ricardian contract solution in the next paragraph.

2.4. Synthesis: Smart Ricardian contracts?

71. Similar to smart contracts, the notion "Ricardian contracts" has been around for longer than what is perceived by the general belief. It was first brought to life by Ian GRIGG in his contribution "Financial Cryptography in 7 layers".[153] GRIGG defined the Ricardian contract as follows:

> *a digital contract that defines the terms and conditions of an interaction, between two or more peers, that is cryptographically signed and verified. Importantly it is both human and machine readable and digitally signed* (emphasis added)

72. The notion Ricardian contract refers to the document which records on the one hand the intentions of one or more contracts, which are yet to be agreed upon and/or executed, and on the other hand the actions which should be taken relating to these contracts. In another contribution, GRIGG clarifies the scope of Ricardian contracts by means of an example.[154] A smart contract designing a crowd funding platform might be an efficient system to allow parties to contribute funds. Participants willing to contribute can deposit cryptocurrency value into the smart contract. When the total amount of funding has reached a predetermined threshold during the stated close time, the smart contract code will invest the funds in the proposed crowdfunding project. If the funds are insufficient to reach the threshold, they will automatically be channelled back to the individual contributors. Escrow agents, bank accounts, and paper become redundant by using such computer protocol instead. Nonetheless, a smart contract setup contains boundaries with respect to the intent of the funding parties. As the smart contract cannot convey the scope of the crowdfunded project, nor the outcome of a possible failure of the project, parties are left with remaining uncertainties. Semantics are missing to clearly indicate the direction of the project and align parties' interests. Ricardian contracts, providing a clear overview of

---

[153] I. GRIGG, "Financial cryptography in 7 layers", *Ian Grigg*, 2000, http://iang.org/papers/fc7.html (Consulted on 20 March 2019).
[154] I. GRIGG, "On the intersection of Ricardian and Smart Contracts", *Ian* Grigg, 2017, http://iang.org/papers/intersection_ricardian_smart.html#ref_Szabo (Consulted on 22 March 2019).

the rationale of parties for engaging in such transactions, might provide an answer to smart contract shortcomings.

73. The central premise of Ricardian contracts remains to be a configuration based on written clauses. Moreover, semantics are key for Ricardian contracts. Again, there is no clarity on the binding character of Ricardian contracts. We propose a pragmatic solution as to whether or not Ricardian contracts are binding. Ricardian contracts can be compared to memoranda of understanding (i.e. MOU's). MOU's are agreements which solely express the intentions of parties with regard to future actions and agreements. Similar to MOU's, Ricardian contracts do not imply any legal commitment, and are therefore not binding. They only express the intentions of parties creating a smart contract.

74. In addition to the semantics in the document which state the intentions of parties, machine-readable tags are attached.[155] These tags are classified as the parameters of the contract, which carry party-specific information, such as price, identity and beneficiary. As an illustration, following excerpt from a Ricardian contract is given, referring to the crowdfunding example which was mentioned before[156]:

```
I, {from}, intent to send {value} to {beneficiary} for the purpose of funding
project {name and function project}.
```

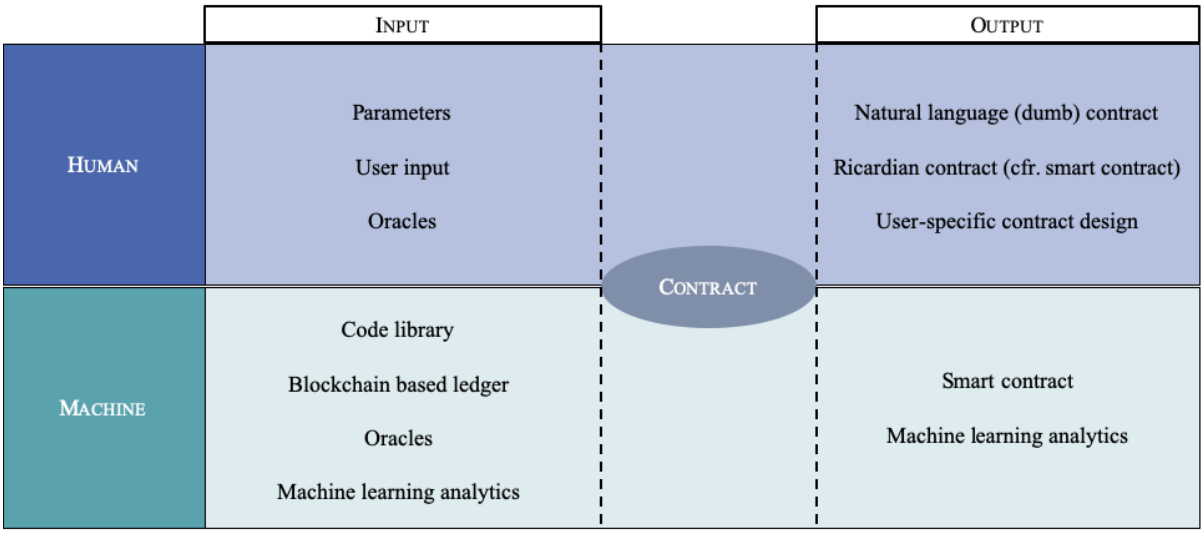
As made clear by the example above, the parameters (i.e. the words between brackets) are shared between the Ricardian contract and the related smart contract.[157] Such constellation safeguards the transparency, which smart contracts often lack. Even though the smart contract will precede the Ricardian contract in terms of legal value, the Ricardian contract might serve as a guideline for further interpretation. In Title V, we will discuss the role of expert panels as oracles. It is important to note that these Ricardian contracts might contribute to a better understanding of the agreement of parties in case disputes arise.

---

[155] U.W. CHOHAN, *What is a Ricardian contract?*, Canberra, University of New South Wales, 11 December 2017, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3085682 (Consulted on 20 March 2019).
[156] For more, see https://eostoolkit.io/
[157] I. BASHIR, *Mastering Blockchain: Distributed ledger technology, decentralization, and smart contracts explained*, Birmingham, Packt Publishing Ltd, 2018, 268.

2.5. Contracts for machines and humans: a new contracting model

75. In this chapter, several paradigm shifts towards a new generation of contracting practices were explained. Both user-centred contracts and machine-centred contracts were mentioned as opportunities towards a new contracting model. In order to provide an answer to the central research question of this contribution, being the question whether or not contract clauses can be translated into smart contract clauses, contracts should be analysed as a whole. Clauses which do not qualify for such translation will need to be adopted in a different manner, to safeguard the comprehensiveness of contracts. Therefore, in the following paragraph, we will attempt to create a contracting model which integrates aforementioned opportunities into a more transparent and efficient hybrid contracting model, as compared to the traditional contracting model based on rather complicated natural language clauses.
Such model should establish an interplay between several platforms and stakeholders. Both humans and machines are involved in the process of drafting, storing and executing the contract. The contract itself forms a hybrid instrument, containing both code and natural language to optimize efficiency. Hence, a distinction is made between the input and output of both humans and machines.
Such new contracting model will be used as a benchmark in Title V to assess the ability of contract clauses to be transformed in smart contract clauses. Moreover, the examined clauses will be integrated into the presented contract model.



a. Input

i. Human input

76. Parties willing to enter a hybrid contract setup are required to provide certain user-specific details according to the predetermined parameters of a certain contract. The adjustment of these parameters might be accomplished through a graphic interface. These parameters correspond with (smart) contract templates, stored in the code library. For example, a smart contract setup for the online second-hand sale of a laptop will require parties to insert their identity, the price, the delivery time and the laptop specifications.[158] Consequently, the corresponding smart contract clauses will be added to the contract. Specifications beyond these parameters can be adopted in the natural language contract. Finally, human-based oracles can be linked to the contract, bridging the gap between the digital existence of smart contracts and real world activity. Human-based oracles can be either be one or more contracting parties, or a trusted third party. We will elaborate on the risks and opportunities of these oracles in Title V.

ii. Machine input

77. From the perspective of the machine, the code library will provide the necessary smart contract templates, optional insertable clauses, programming language facilities and code compilers.[159] The code library is linked to the parameters of the contract, as explained before. A decentralised ledger will form the supporting surface for the contract, ensuring its tamper-proof self-executable character. Machine-based oracles can be connected to the contract to provide details on real world events, such as sensors identifying the delivery of certain goods. Lastly, machine learning analytics can be incorporated to provide suggestions to the contracting parties or determine a reasonable price.[160] Hence, to a greater extent, consistency and clarity will be achieved across the contracts which are concluded by a business.

b. Output

78. The output side is characterised by an internal smart contract model. Both the natural language (dumb) contract and the smart contract are on the same level in terms of legal value.

i. Human output

79. As a result, a binary output is generated and separated into two channels. The first output channel is aimed towards a human interpretation of the contract. This includes a so-called "dumb" contract or natural language contract, in which non-operational clauses are adopted, such as a confidentiality clause.[161] In addition, the Ricardian contract is displayed to clarify the correlated smart contract's intentions.[162] Finally, by means of a graphically designed representation of the contract, every interested party receives an individualised overview of the contract.[163] When combined with contract management software, these graphic representations might enhance the user's knowledge and experience of the concluded contracts.

---

[158] For an actual example of an operating platform, see https://openbazaar.org. OpenBazaar does not charge any transaction fees, as opposed to conventional platforms such as eBay.
[159] *Infra*, Title I.TITLE V.1.2.
[160] *Supra,* 57.
[161] *Infra*, 161.
[162] *Supra,* 73.
[163] An example can be the automated graphic representation of insurance contracts for the board of directors of an insurance company. Such representation can eliminate all excess information and extract relevant aspects of a number of contracts, providing a more efficient overview; See *Infra,* 89.

ii. Machine output

80. The second output channel is aimed towards a machine-based interpretation of the contract. This model opts for an internal smart contract approach, explicitly implying the legal value of the code embedded in the smart contract.[164] The smart contract part resides on the distributed ledger, accompanied by its related Ricardian contract. In doing so, the uniformity between the two contracts is ensured. Finally, machine learning tools can be applied to further develop the algorithm behind the decentralised ledger. Potential fraudulent behaviour can be discovered, or price adjustment recommendations might be suggested by the algorithm.

81. The aforementioned contract model indicates the importance of an exhaustive overview of which types of contractual clauses can be adopted by smart contracts. Else, these clauses should be externalised in the natural language contract. Title V will attempt to delineate those clauses which are impossible to be drafted in a coded form, mainly focussing on commercial contracts.

[164] *Supra*, 68.

CHAPTER 3. OPPORTUNITIES FOR SMART CONTRACTS

82. This chapter will illustrate some of the opportunities smart contracts created. We should point out that the selection of these cases is based on blockchain based smart contracts, as opposed to the broad category of smart contracts.

The nature of smart contracts both limits and increases potential use cases. Therefore, it is essential to assess which industries might be affected by the implementation of this new technology. The online environment in which smart contracts reside, implies certain content restrictions on the types of contracts which can be structured as smart contracts. For instance, service contracts based on the personal performance of one of the contracting parties will not qualify for smart contract uses, due to the impossibility of self-execution of a personal obligation. Besides considerations regarding the possibility of smart contract implementation, one should analyse the efficiency benefits such implementation might bring. We will give a summarized overview of the industries which might benefit from smart contracts, giving examples of the undertaken initiatives.

3.1. Use cases

a. Financial transactions

83. Evidently, financial incentives set the ball rolling towards other applications for smart contracts. As explained above, Bitcoin transactions are considered prototype smart contracts, admittedly with limited functionality. In the Ethereum Turing-complete environment, a vast amount of different financial transactions can be realised. Their desirability is backed by statistics depicting the rising problems with traditional financial contracts.[165] Especially in the financial industry, trust is a valued precondition upon entering into transactions with third parties. A lack of trust imposes a substantial economic cost on the financial market. Major financial institutions have been examining the opportunities of smart contracts and several initiatives were launched to exploit the technology and remedy these trust-related issues. An example is Quorum, a platform launched by JP Morgan Chase, which establishes a permissioned blockchain and enables private smart contract execution.[166]

[165] For example, the London diamond industry is confronted with a two billion dollar cost of fraud each year ; B. CANT, A. KHADIKAR, A. RUITER, J. BRONEBAKK, J. COUMAROS, J. BUVAT en A. GUPTA, "Smart contracts in financial services: getting from hype to reality", *Capgemini Consulting*, 2016, 4, https://www.capgemini.com/consulting-de/wp-content/uploads/sites/32/2017/08/smart_contracts_paper_long_0.pdf (Consulted on 26 February 2019).
[166] For more, see https://www.jpmorgan.com/global/Quorum

84. One of the many useful applications is a smart contract that hedges against the volatility of cryptocurrencies in relationship to a non-virtual currency, such as the Euro.[167] This is often realised by using options and futures. The former gives the owner the right to buy (call option) or sell (put option) crypto assets at a predetermined price.[168] The latter refers to the obligation to buy or sell a crypto asset at a predetermined price.[169] As opposed to an option, futures leave no margin for appreciation for the owner, as the transaction is not optional. Futures and put options are a commonly used financial derivative to hedge the risk of volatility and can be realised in smart contract form. For instance, when a certain crypto asset value is hit, the contract executes itself and buys or sells the assets.

A second example is based on the capacity of smart contracts to hedge the risk against non-buyer payment.[170] One of the implications of smart contracts could be the decentralisation of escrow services.[171] In a traditional financial escrow model, escrow agent services reassure payments in a relationship lacking trust. For example, when a seller decides to sell his phone online, the buyer might be reluctant to transfer the purchase price prior to having received the phone. This is often due to the absence of trust. Therefore, the money is first transferred to an escrow agent. The agent will release the funds when the buyer confirms he has received and accepted the delivered phone.[172] However, certain service fees are taken by the escrow agents. Also, the escrow agent services are prone to the risk of insolvency or fraud. Smart contract might constitute a similar escrow system, while increasing the security and transparency of the transaction. Parties no longer need to rely on a trusted middleman to release the funds, as smart contracts operate independently. Prior to the transaction, the seller will insert cryptocurrency value in the smart contract escrow that is worth twice the value of the good he is planning to sell *2V (= 2 times the value*). The buyer then confirms the transaction by inserting the same value *2V*. Once the good reaches the buyer, he will trigger the smart contract to execute and confirm the delivery. *3V* returns to the seller, which consists of a purchase price of *1V* and the initially inserted

---

[167] V. BUTERIN, "A next-generation smart contract and decentralized application platform", *supra* from 7, 2014, 20.

[168] J. HULL, S. TREEPONGKARUNA, D. COLWELL, R. HEANEY en D. PITT, *Fundamentals of futures and options markets*, Australia, Pearson Higher Education AU, 2013, 7.

[169] *Ibid,* 2.

[170] J.S. GANS, "The Fine Print in Smart Contracts", *SSRN*, 3 January 2019, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3309709 (Consulted on 4 March 2019).

[171] A. BOGNER, M. CHANSON en A. MEEUW, *A decentralised sharing app running a smart contract on the ethereum blockchain*, ACM, Proceedings of the 6th International Conference on the Internet of Things, 2016, http://cocoa.ethz.ch/downloads/2017/08/2306_Sharing_App_Final_Publication.pdf (Consulted on 25 March 2019).

[172] An example of an escrow agent service is Simplex. For more, see https://www.simplex.com/escrow-service/.

stake of *2V*. The remaining *1V* are sent to the buyer and is intended to incentivise him to confirm the deliverance of the good.[173] Systems exploiting this escrow service are already existent.[174]

b. Internet of Things and smart property

85. The Internet of Things establishes an ever growing interconnectivity between people and devices.[175] Ordinary devices, such as doorknobs or lightbulbs, will have the capacity of sending and receiving data to other devices. The number of connected devices is forecasted to surpass 25 billion in 2020, making visible the scale of opportunity for the Internet of Things.[176] However, businesses face several challenges in adapting their business models to commercialise Internet of Things applications. One of these challenges is the substantial cost associated with installing and maintaining large centralised data centres.[177] Moreover, from the customer's side, there remains a lack of trust with respect to connected devices in a home setting.[178] A decentralised peer-to-peer network can effectively resolve these issues and increase scalability, since it is possible to combine computational power on the one hand and store data on millions of interconnected devices, which are sitting idle most of the time, on the other hand.[179] Furthermore, establishing a decentralised blockchain network facilitates smart contract opportunities. For instance, Slock.it created a blockchain based application which users can use to set up a smart contract for rental purposes. Through the use of smart locks, real-world objects can be controlled according to predetermined instructions embedded in a smart contract. These smart locks or "slocks" represent the access rights to houses, cars, bikes, etc. In order to unlock them, users need to download the Slock app and make a payment (in Ether) which covers rental costs and a deposit amount. Upon returning the object, the deposit is returned to the owner.[180]

c. Intellectual property rights

86. Intellectual property rights are related to creations of human intellect. These creations, either artistic or technical, are characterised by a non-physical existence. Therefore, in order to commercially exploit those creations, the law attaches certain rights to these intangible ideas. Due to their immaterial nature, the registration of their existence requires some sort of ledger, which determines the authenticity

---

[173] J. NG, "Escrow Service as a Smart Contract: The Business Logic", *Medium*, 20 May 2018, https://medium.com/coinmonks/escrow-service-as-a-smart-contract-the-business-logic-5b678ebe1955 (Consulted on 25 March 2019).
[174] An example is Safe Remote Purchase. For more, see https://solidity.readthedocs.io/en/develop/solidity-by-example.html#safe-remote-purchase.
[175] *Supra*, 48.
[176] IBM INSTITUTE FOR BUSINESS VALUE, "Device democracy: Saving the future of the Internet of Things", IBM, 2014, https://www.ibm.com/downloads/cas/Y5ONA8EV (Consulted on 14 March 2019).
[177] *Ibid.*
[178] K. CHRISTIDIS e.a., "Blockchains and smart contracts for the internet of things", *supra* from 111, 2016, 2298.
[179] IBM INSTITUTE FOR BUSINESS VALUE, "Device democracy: Saving the future of the Internet of Things", *supra* from 194, 2014.
[180] For more, see www.slock.it

and ownership of the rights. Establishing a blockchain network to facilitate a decentralised ledger might generate additional advantages over a centralised ledger. Examples of use cases include: evidence of creatorship and provenance authentication, registering and clearing IP rights; controlling and tracking the distribution of (un)registered IP; providing evidence of genuine and/or first use in trade and/or commerce; digital rights management (e.g., online music sites); establishing and enforcing IP agreements, licenses or exclusive distribution networks through smart contracts; and transmitting payments in real-time to IP owners.[181] The example of Ujo Music, a company using smart contracts to license and commercialise musical content, was already mentioned above.[182] Another example is KODAKCoin, a new platform built by Kodak. Kodak has been an industry leader in the production of photographic cameras. As photographers all around the world are confronted with the unlawful exploitation of their art, and control has become increasingly difficult, Kodak introduced a blockchain network to register and license copyrights of any professional or amateur photographer. Furthermore, the platform acts as a web crawling tool, monitoring the legitimate and illegitimate exploitation of registered copyrights.[183]

d. E-voting and share/token exchange

87. Similar to the double spending problem that prevented cryptocurrencies from being tamper-proof, E-voting mechanisms are confronted with a "double voting" issue. Several E-voting structures have been tested, but concerns remain with regard to the reliability and the transparency of such structures.[184] Estonia was one of the first countries to effectively introduce an online comprehensive voting system. The government distributed smart digital ID cards and personal card readers to enable voter authentication.[185] By entering a web portal, remote votes can be cast by anyone that has access to an Internet connection. Furthermore, systems are implemented which allow citizens to publicly propose a legislative act and, as a consequence, are allowed to gather supporting citizens to digitally sign these proposals.[186] Even though such constellation is a prime example of technology embracing and

---

[181] B. CLARK, "Blockchain and IP Law: A Match made in Crypto Heaven?", *Wipo Magazine*, February 2018, https://www.wipo.int/wipo_magazine/en/2018/01/article_0005.html (Consulted on 16 March 2019).
[182] *Supra,* 47.
[183] P. BAJPAI, "Kodak, The Blockchain and Cryptocurrency: How Kodak Is Tapping Into Technology", *Nasdaq*, 28 January 2018, https://www.nasdaq.com/article/kodak-the-blockchain-and-cryptocurrency-how-kodak-is-tapping-into-technology-cm911406 (Consulted on 16 March 2019).
[184] A.K. KOÇ, E. YAVUZ, U.C. ÇABUK en G. DALKILIÇ, "Towards secure e-voting using Ethereum blockchain", *Proc. 6th International Symposium on Digital Forensic and Security (ISDFS 2018)*, 2018, 2, www.researchgate.net/profile/Umut_Cabuk/publication/323318041_Towards_Secure_E-Voting_Using_Ethereum_Blockchain/links/5a931e2faca272140565c7e4/Towards-Secure-E-Voting-Using-Ethereum-Blockchain.pdf (Consulted on 17 March 2019).
[185] E. MAATEN, "Towards remote e-voting: Estonian case", *Electronic Voting in Europe-Technology, Law, Politics and Society*, 2004, 47, 85, https://pdfs.semanticscholar.org/ff4d/0a77e7561e62fd0258280c0baa02d8256a03.pdf (Consulted on 17 March 2019).
[186] A.K. KOÇ e.a., "Towards secure e-voting using Ethereum blockchain", *supra* from 184, 2018, 2.

supporting democratic legitimacy, the Estonian voting mechanisms are still prone to hacking and hijacking attempts. This results from the centralised configuration of the administration of the votes. Not only governmental voting facilities are affected by these E-voting developments. In corporate structures, shareholders are required to identify themselves upon exercising their shareholder rights (i.e. their voting right). An increasing amount of shareholders already cast their vote electronically.[187] Establishing a permissioned distributed ledger, which would incorporate a set of rules for shareholder voting, will reduce the number of intermediaries (such as central security depositories) and reduce the risk of the voting results being hacked.[188] Every voter would receive a copy of the voting report, being a node in the network, which leads to an improved level of transparency. Furthermore, smart contracts could be deployed on top of this network, functioning as shareholder agreements to automatically take action under certain conditions. For instance, in case of specific voting results, certain investment decisions are made by the company.[189] At present, different blockchain based voting systems are being tested, aiming to guarantee maximum voter privacy.[190]

88. Similar to an initial public offering (IPO) (i.e. the operation in which a company is listed on a stock exchange and sells its shares to the public), several companies are now experimenting with an ICO, or an initial coin offering. In such case, the company issues coins, or tokens, which entitle the owner of the coins to certain rights. These tokens can secure voting and can allocate rights to the holder with regard to the usage of certain assets of the company, such as software.[191] Such token sales are realised through the use of smart contracts.[192] These coins can be sold through smart contracts to interested parties. Since 2017, blockchain start-ups have raised over 7 billion dollar through initial coin offerings, demonstrating the attention which is devoted to such alternative venture capital structures by the industry.[193]

---

187 C. VAN DER ELST e.a., "Bringing the AGM to the 21st Century: Blockchain and Smart Contracting Tech for Shareholder Involvement", *supra* from 103, 2018, 11.

188 *Ibid,* 20.

189 EUROPEAN UNION, "What if blockchain technology revolutionised voting?", *At a glance*, 2016, http://www.europarl.europa.eu/RegData/etudes/ATAG/2016/581918/EPRS_ATA%282016%29581918_EN.pdf (Consulted on 18 March 2019).

190 P. MCCORRY, S.F. SHAHANDASHTI en F. HAO, "A smart contract for boardroom voting with maximum voter privacy", *International Conference on Financial Cryptography and Data Security*, 2017, 2, https://www.researchgate.net/publication/317843497_A_Smart_Contract_for_Boardroom_Voting_with_Maximum_Voter_Privacy (Consulted on 18 March 2019).

191 Y.-Y. HSIEH en J.-P. VERGNE, "Bitcoin and the rise of decentralized autonomous organizations", *Journal of Organization Design*, 2018, vol. 1, 9.

192 K.D. WERBACH, "Trust, but verify: Why the blockchain needs the law", *supra* from 46, 2018, 518.

193 C. CATALINI en J.S. GANS, "Initial coin offerings and the value of crypto tokens", National Bureau of Economic Research, 2018, 1, https://www.nber.org/papers/w24418 (Consulted on 2 May 2019).

e. *Insurance*

89. The insurance industry has been investigating blockchain opportunities for a couple of years now. The intrinsic nature of insurance policies often relates to simple "if-then" statements, making those contractual relationships an excellent candidate for optimisation through the use of code (e.g. if person X passes away, then life insurance is paid). From the perspective of customer experience, smart contracts could realise faster policy fee payments to policy-holders, as predetermined payment triggers can be adopted in the smart contract.[194] Also, because of the oracle data feed into the blockchain, there's only one source of information to rely on. Hence, the risk of fraudulent claims is reduced. It should be noted, however, that insurance companies align their business model with mathematic prediction models. This implies that, in order to determine the insurance premium and the amount of insurance payment, accurate predictions are indispensable. Therefore, it is likely to see smart contract integration in rather short-term and/or less complex insurance relationships. For instance, current life insurance policies require survivors to provide a death certificate or a proof of loss. Multiple issues may arise for the survivors, such as the unawareness of the existence of a life insurance policy or the disappearing of the necessary documents.[195] A blockchain based solution would resolve both aforementioned issues: the contract would automatically execute with the policyholder's decease, triggered by an oracle input. Furthermore, it is nearly impossible for the policyholder to permanently lose his policy, as the policy is retained with every node in the network.

Other opportunities can be found in the field of parametric insurance. Pay-outs are based on objective measures, as opposed to a claims adjuster surveying damages.[196] These objective measures can be generated by weather reports or catastrophe evaluations. Often, parametric insurance is used to remedy acute distress situations. In case of a heavy storm, the affected victims are not ought to wait several months for their insurance pay-outs to initiate repair. A parametric insurance smart contract can be programmed to obtain information relating to the size and conditions of the storm in order to determine whether the policy is activated.[197]

An example of an operating insurance facility through the use of smart contracts is Fizzy, created by French insurance giant AXA. Smart contracts are linked to global air traffic databases. Functioning as an oracle, these databases will share flight delay times with the smart contract. As soon as a delay of more than two hours is registered, compensation is automatically triggered.[198]

---

[194] V. GATTESCHI e.a., "Blockchain and smart contracts for insurance: Is the technology mature enough?", *supra* from 104, 2018, vol. 2, 6.

[195] A. COHN, T. WEST en C. PARKER, "Smart after all: Blockchain, smart contracts, parametric insurance, and smart energy grids", *Georgetown Law Technology Review*, 2017, vol. 2, 291.

[196] *Ibid,* 293.

[197] M. GIANCASPRO, "Is a 'smart contract' really a smart idea? Insights from a legal perspective", *supra* from 7, 2017, vol. 6, 834.

[198] For more, see https://fizzy.axa/fr/

f. Supply chain

90. Supply chain is an umbrella term for the lifecycle containing people and things that are involved in bringing products from the place of production to the place of the buyer. The trajectory usually covers a variety of stages in which several economic entities play a role in getting the product to the customer. Hence, there's a significant amount of information exchange, transaction costs and risks of demise of the goods along the track. Evidently, coordination between multiple producers, suppliers and distributors requires a vast amount of paperwork, ranging from purchase orders to shipping confirmations to proof of receipts.[199] Parties act in their own best interests and negotiate terms and conditions of each contract separately, leading to a phenomenon that is called "battle of the forms".[200] Keeping track of such a massive information flow is the principal challenge to overcome in most supply chains. Attempts to synchronise every link in the chain are costly. Another evolution to be aware of is the growing desire of consumers to identify and appreciate the trajectory a certain product has completed before ending up in a retail store. Modern-day perspectives no longer tolerate the production of clothing made by children or underpaid workers.

91. Centralised ledgers are considered ineffective to answer aforementioned concerns, because of the conflicting interests of the parties participating in the supply chain network. There's an intrinsic lack of trust between the participants of the network, as one of the parties might tamper with the centralised ledger and manipulate the embedded information. However, such lack of trust can be remedied by a blockchain network, which is the optimal solution for a group of entities that have a common goal but do not fully trust each other.[201] In doing so, the possibility of deploying smart contracts is enabled. The synergy between smart contracts on the one hand and the internet of things on the other hand can streamline certain distribution operations. Especially in the industries that impose strict requirements with regard to transportation conditions and maintenance of the goods, the use of data generating sensors can contribute to more transparency. For instance, the pharmaceutical supply chain has many complex environmental control processes which require the goods to be transported in predetermined temperature and humidity levels.[202] There's an obligation for the distributor to monitor these parameters at all times.[203] Smart contracts would reduce the number of intermediaries, while

---

[199] B. BHANDARI, *Supply Chain Management, Blockchains and Smart Contracts*, New York, NYU University of Law, 2018, 3, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3204297 (Consulted on 18 March 2019).
[200] *Ibid,* 4.
[201] Such blockchain ledger might be a permissioned ledger. In such case, users are subjected to the condition of authorisation by a central authority or consortium; E. ANDROULAKI e.a., "Hyperledger fabric: a distributed operating system for permissioned blockchains", *supra* from 41, 2018, 1.
[202] T. BOCEK, B.B. RODRIGUES, T. STRASSER en B. STILLER, "Blockchains everywhere-a use-case of blockchains in the pharma supply-chain", *2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*, 2017, 773, http://dl.ifip.org/db/conf/im/im2017exp/119.pdf (Consulted on 20 March 2019).
[203] Guidelines (Comm.) of 5 November 2013 on Good Distribution Practice of medicinal products for human use, *Official Journal of the European Union,* 23 November 2011, 2013/C 343/01 ; Article 84 and Article 85b(3) of

maintaining compliance with the temperature and humidity requirements.[204] By using sensors, a live data feed is set up throughout the entire process of transportation of the medicinal products. Any deviations from the predetermined parameters will be visible for the participants of the underlying permissioned blockchain network. Furthermore, smart contracts can attach legal consequences to the non-compliance of one of the links in the supply chain, stipulating price reductions or exceptions of non-performance. We will elaborate on supply chain possibilities in Title V.

92. Several start-ups exploit these new opportunities by offering tailored service models to the industry. The aforementioned example of medicinal product supply chain is targeted by Modum.io. The Zurich-based start-up successfully completed a pilot project, monitoring 55 shipments, and currently cooperates with major technology partners such as SAP.[205] Another example is T-Mining, a Belgian-based start-up which provides blockchain solutions in the Port of Antwerp. They aim to enhance international container transport efficiency through the application of multiple improvements. For instance, T-Mining developed a decentralised system for secure and efficient document workflow. At present, documents related to the shipping of containers come in many shapes and sizes, as there are many intermediaries playing a role in the transportation. On average, 30 companies are involved in such transportation, leading to complex relationships and high opacity.[206] T-Mining offers a more transparent solution which allows for the transfer of rights in a paperless and more efficient way and guarantees the authenticity of documents. Smart contracts determine the release of certain documents, therefore transferring the ownership and the risk of the cargo, when predetermined criteria are met.[207]

3.2. Disruption?

93. Even though plenty of use cases can be distinguished, a true revolution has yet to be identified. Scepticism is omnipresent towards the commercial viability of smart contracts on a large scale.[208] Evidently, the worldwide adoption of internet did not happen overnight. It often takes a handful of pioneering entrepreneurs or corporations to introduce an attractive business model that exploits these

---

Directive 2001/83/EC of the European Parliament and of the Council of 6 November 2001 on the Community code relating to medicinal products for human use, *Official Journal of the European Union*, 28 November 2011.

[204] T. BOCEK e.a., "Blockchains everywhere-a use-case of blockchains in the pharma supply-chain", *supra* from 202, 2017, 776.

[205] *Ibid,* 776; For more, see https://modum.io

[206] As pointed out by N. WOUTERS during his lecture "Blockchain 101 – Technische aspecten van blockchain", Leuven, 21 June 2018.

[207] For more, see https://t-mining.be/news1/2018/6/12/antwerp-blockchain-pilot-pioneers-with-secure-and-efficient-document-workflow

[208] K. O'HARA, "Smart contracts - Dumb idea", *supra* from 10, 2017, vol. 2, 100; D. GERARD, *Attack of the 50 foot blockchain: Bitcoin, blockchain, Ethereum & smart contracts*, USA, CreateSpace Independent Publishing Platform, 2017; E. TJONG TJIN TAI during his lecture "Blockchain 101 – Technische aspecten van blockchain", Leuven, 21 June 2018.

new technologies. This paragraph will concisely summarize the remaining legal and non-legal challenges.

a. Legal challenges

94. The confrontation of smart contracts with the existing body of contract law does not result in a perfect match. For centuries, contracting practices have been an engine towards the democratic development of contract law.[209] Regulation often counters undesired market behaviour, which subsequently leads to the adaptation of such behaviour. One can argue that the established regulation on contract practices reflects an ever-evolving society. Smart contracts shed a new light on these settled doctrines, as they are a radically different version of the contracts to which society is accustomed to. Among others, following concerns are raised with regard to the legality of smart contracts.

i. Establishing capacity[210]

95. Establishing capacity refers to the ability of a party to enter into a contract. Several aspects can affect whether or not a party is considered capable of entering into a contract under a legal system. One of these aspects is the age of the parties. In Belgium, a person aged 18 can legitimately conclude a contract.[211] Most legal systems contain several exceptions on the fundamental establishing capacity of adults, such as a lack of competency. A person contracting without legitimate establishing capacity creates a contract which is voidable.[212] Often in a decentralised network, the counterparty cannot be aware of a party's incapacity because both parties might be unknown to each other. As a consequence, the nullity of a contract imposes a restitution duty on both parties. This possibility is at odds with the ability of automated execution of smart contracts, as immediate remedies by judicial recourse are rendered impossible. However, such threats might be mitigated by using electronic identification systems. In doing so, parties will need to identify themselves and prove their capacity before a smart contract can be concluded.

ii. Formation[213]

96. The traditional theory of offer and acceptance of a contract is also applicable to smart contract formation. Belgium applies the doctrine of "the meeting of the minds" to determine whether or not a contract is concluded.[214] However, one should be aware of the exceptional circumstances under which these contracts are formed. Belgian legislation, just like French legislation, does not specify any

---

[209] E. TJONG TJIN TAI, "Formalizing Contract Law for Smart Contracts", *supra* from 4, 2017.
[210] M. GIANCASPRO, "Is a 'smart contract' really a smart idea? Insights from a legal perspective", *supra* from 10, 2017, vol. 6, 825–835.
[211] Art. 388 of the Belgian Civil Code ; Art. 1124 of the Belgian Civil Code.
[212] Art. 1125, section 2 of the Belgian Civil Code.
[213] M. RASKIN, "The Law and Legality of Smart Contracts", *supra* from 14, 2017, vol. 2, 322-326.
[214] Art. 1101 of the Belgian Civil Code.

particular indications as to when the acceptance of an online contract takes place. In the case where parties communicate and negotiate before the creation of the smart contract, there's no ambiguity with regard to the moment of acceptance.[215] However, in the situation where such communication does not occur, one can argue that the moment of acceptance takes place when the counterparty undertakes action to initiate this acceptance. More specifically, in some cases such action might be the actual trigger to execute the smart contract.[216] Acceptance and execution of the contract occur almost simultaneously. At present, legal systems are not adapted to such possibilities. This might lead to more complex conflicting situations, for instance when courts apply the doctrine of trust. The doctrine of trust (*vertrouwensleer*) is applied by Belgian courts to remedy a discrepancy between the externalized will and the actual will of a contracting party. Such a discrepancy might lead to the annulment of the contract.

iii. Interpretation

97. According to article 1108 of the Belgian Civil Code, a valid contract must have a legitimate object (*voorwerp*). In order to be legitimate, an object must be reasonably determinable. However, as smart contracts as such are only legible for trained professionals, it is nearly impossible for a court to assess the legitimacy and the certainty of the object of the contract.[217] Such an impasse gives rise to a complicated issue of interpretation when a disagreement between the parties occurs. The courts might rely on pre-contractual documents to assess the intentions of the contracting parties, but these prefatory documents do not contain any legal value on their own and, therefore, cannot be considered to replace the smart contract for the purpose of interpretation. Courts might have recourse to programmers in the capacity of court experts, who can decipher the smart contract and reflect their objective translation. However, courts will be bound to form their judgement based on an external interpretation of the contract, leading to possible violations of the rights of the defence.

iv. Performance and modification[218]

98. Contracting parties can be confronted with unanticipated circumstances during the term of the contract. Several legal doctrines provide an excuse for breach of contract. The Belgian legal system adopted the theory of force majeur.[219] In the event of unforeseen circumstances that impede the performance of one or more contractual obligations, force majeur might be invoked by the debtor as an excuse for non-performance. However, certain conditions should be met to justify such non-performance. The event should escape the control of the debtor (*onvoorkomelijk*) and should not be

---

[215] M. GIANCASPRO, "Is a 'smart contract' really a smart idea? Insights from a legal perspective", *supra* from 7, 2017, vol. 6, 830.
[216] M. RASKIN, "The Law and Legality of Smart Contracts", *supra* from 14, 2017, vol. 2, 322.
[217] M. GIANCASPRO, "Is a 'smart contract' really a smart idea? Insights from a legal perspective", *supra* from 7, 2017, vol. 6, 831.
[218] M. RASKIN, "The Law and Legality of Smart Contracts", *supra* from 14, 2017, vol. 2, 326.
[219] Article 1147 of the Civil Code; Article 1148 of the Civil Code.

reasonably be foreseen at the conclusion of the contract (*onvoorzienbaar*).[220] The self-executable character of smart contracts pose a new challenge for the settled force majeur doctrine. The debtor will not be able to invoke force majeur prior to the execution of the contract, because the contract executes itself without judicial recourse. The computer protocol supporting the smart contract is not familiar with the notion of force majeur. Smart contracts require an anticipating approach towards remedying non-performance of one of the parties. This quality is both its weakness and its strength. Contract law is about ex post adjudication, which is diametrically opposed to the ex ante character of smart contracts.[221] Title V will examine how this apparent contradiction can be resolved.

99. As mentioned before, one of the fundamental characteristics of a blockchain network is its immutability.[222] As a consequence, smart contracts embedded in such a network are immutable as well.[223] However, during the course of operation of smart contracts, several aspects might be wished by parties to be modified. Smart contracts should therefore include an option to breach their immutability, for example to be adapted to new legislation which would render the contract illegitimate.[224] This modification possibility would be at odds with one of the major selling points of smart contracts: the guaranteed execution of the clauses that parties agreed upon.[225]

b. Non-legal challenges

100. An extensive analysis of non-legal challenges related to the viability of smart contracts goes beyond the scope of this research. However, it should be noted that, from a technological perspective, remaining hurdles still need to be taken. The value of smart contract applications depends significantly on the reliability and security of such applications. A couple of examples will be described.

101. As mentioned before, the use of oracles might enhance the impact of smart contracts on physical assets.[226] Real world data input can be fed into the decentralised ledger. Nonetheless, there should be adequate guarantees that information from this external source is trustworthy. Oracles do not create or compute information, but process information from external data resources, such as websites or

---

[220] E. TJONG TJIN TAI, "Force Majeure and Excuses in Smart Contracts", *European Review of Private Law*, 2018, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3183637 (Consulted on 15 March 2019); A. VAN OEVELEN en A. DE BOECK, "Overmacht en imprevisie in het Belgische contractenrecht" in *Overmacht*, Mortsel, Intersentia, 2015, 2, https://www.jurisquare.be/en/book/9789400006911/overmacht-en-imprevisie-in-het-belgische-contractenrecht/ (Consulted on 27 March 2019).
[221] E. TJONG TJIN TAI, "Force Majeure and Excuses in Smart Contracts", *supra* from 205, 2018, 18.
[222] *Supra,* 24.
[223] M. GIANCASPRO, "Is a 'smart contract' really a smart idea? Insights from a legal perspective", *supra* from 8, 2017, vol. 6, 826.
[224] M. RASKIN, "The Law and Legality of Smart Contracts", *supra* from 14, 2017, vol. 2, 327.
[225] A. SAVELYEV, "Contract law 2.0: 'Smart'contracts as the beginning of the end of classic contract law", *supra* from 71, 2017, vol. 2, 20.
[226] *Supra*, 47

predictions markets.[227] The use of oracles therefore implies introducing a certain degree of uncertainty back into the system, as they can be manipulated to achieve a certain smart contract outcome.[228]

102. A second example is the so called "DAO-event". DAO stands short for Decentralised Autonomous Organisation. The platform was based on a complex interlinked system of smart contracts and set up a democratic venture capital fund.[229] Applicants seeking funding would present their business plan to the DAO token holders. A democratic vote then would decide whether or not the project would obtain investment funds. The DAO formed the largest crowdfunding campaign in history, as more than 150 million USD was collected.[230] However, a major turnaround took place when an Ethereum user inserted a subtle bug in the contract. The user was able to extract 60 million USD worth of Ether out of the funds to a contract that belonged to her.[231] As a result, the Ethereum user community was split up into two factions. The majority of Ethereum users upgraded Ethereum by performing a hard fork initiated by Vitalik BUTERIN, Ethereum's founder.[232] The hard fork (i.e. a legitimate hack of the system, based on a consensus by the participants of the network) bypassed the hacker's transaction and could return the funds to their corresponding rightful token holders in the DAO.[233] On the contrary, advocates of legal automation rejected such interference, as this would violate the spirit of smart contracts and blockchain. This led to the creation of two separate blockchains: Ethereum (including the rollback of the DAO-hack) and Ethereum classic (including the actual DAO-hack without the hard fork).[234] The DAO-hack showed one of the remaining weaknesses of smart contracts. Not only the technology as a whole has to be tamper-proof, but even more important, the underlying code should resist attacks from malicious users. Furthermore, some argue that if the code and only the code is the contract, the successful operation launched by the hackers is in a sense permitted conduct.[235]

103. The third example resides at the crossroads of law and technology. The prerequisite abilities to write a piece of software are fundamentally different from the necessary skills to draft an appropriate legal contract. Combining these skills requires a thorough comprehension of legal semantics:

---

[227] E. MIK, "Smart contracts: terminology, technical limitations and real world complexity", *Law, Innovation and Technology*, 2017, vol. 2, 296.
[228] P. CUCCURU, "Beyond bitcoin: an early overview on smart contracts", *International Journal of Law and Information Technology*, 2017, vol. 3, 186; J. GRIMMELMAN, "All Smart Contracts Are Ambiguous", *Penn Journal of Law and Innovation*, 2019, 15, https://ssrn.com/abstract=3315703.
[229] K. O'HARA, "Smart contracts - Dumb idea", *supra* from 10, 2017, vol. 2, 98.
[230] *Ibid.*
[231] C. DANNEN, *Introducing Ethereum and Solidity: Foundations of Cryptocurrency and Blockchain Programming for Beginners*, *supra* from 84, 2017, 8.
[232] A fork refers to the process of persuading the entire network of nodes to upgrade, or agree, on a new version of Ethereum.
[233] F.A. PASQUALE, "A Rule of Persons, Not Machines: The Limits of Legal Automation", *George Washington Law Review*, 2018, 39.
[234] J. GRIMMELMAN, "All Smart Contracts Are Ambiguous", *supra* from 113, 2019, 19.
[235] K. O'HARA, "Smart contracts - Dumb idea", *supra* from 10, 2017, vol. 2, 99.

understanding the meaning of legal words on the one hand, and the logical meaning of contract clauses on the other hand.[236] It forces lawyers to adopt a more interdisciplinary approach. Furthermore, lawyers will need to learn how to cooperate with engineers and technology experts.[237] In this regard, another challenge refers to the amount of literalism that should be exercised when drafting smart contracts, compared to drafting traditional contracts. In the case of a rising dispute, courts interpret traditional contracts while applying a certain framework based on a sense of commercial awareness and psychological understanding. Smart contract execution bypasses such a framework. As a result, this should be taken into account prior to the actual judicial execution.

CHAPTER 4. PRELIMINARY CONCLUSION

104. Title IV provided an overview of the 21st century contractual landscape from a technological perspective. Several innovative developments were discussed: on the one hand, evolutions that extend and support the existing traditional contractual model, such as artificial intelligence integration, and on the other hand, evolutions that disrupt the contemporary contracting practices, such as contract design. We can conclude that contracts can benefit from technology, and that there's still room for improvement.

Moreover, we gave an overview of the different smart contract models (internal vs external smart contract model). An internal smart contract model, combined with Ricardian contracts is considered the right option for further smart contract development. Such option supposes that smart contracts contain the same legal value as their corresponding natural language contracts. To resolve the issue of interpretation, we suggest the adoption of Ricardian contracts, expressing the intentions of the parties that draft the smart contract.

105. As a result, a new contracting model was proposed, integrating such evolutions to abandon the established perception of contracts "written by lawyers, for lawyers". The contract model, containing two output channels, aims to provide contracts for both humans and machines on the one hand, and integrate modern technologies, such as AI on the other hand. Smart contracts play a prominent role in aforementioned model, forming the automated machine-readable side of the contract.

106. Subsequently, chapter 3 illustrated potential use cases for blockchain based smart contracts. We can conclude that there are interesting opportunities for smart contracts, in particular in the industries of insurance, supply chain and finance.

---

[236] C.D. CLACK, "Smart Contract Templates: Legal semantics and code validation", *Journal of Digital Banking*, 2018, vol. 4, 1.

[237] M. RASKIN, "The Law and Legality of Smart Contracts", *supra* from 14, 2017, vol. 2, 325.

Even though plenty of commercial applications can be identified, a considerably disrupting exploitation of smart contracts has yet to be launched. Up until now, the overall impact on the contractual landscape has been rather modest. This might be caused by remaining legal challenges, such as the contested (in)ability of concluding a valid contract in smart contract form. In addition, certain non-legal challenges need to be overcome as well.

Acknowledging the long road ahead, a first evaluation leads to the conclusion that an integration of smart contracts is certainly not impossible. As mentioned before, operation costs of business transactions might be reduced significantly by the implementation of smart contracts.[238] We believe market forces are the driving force behind overcoming legal and non-legal challenges.[239] Those same market forces led to the integration of the internet, smartphones and many more disruptive products of technological development. Similar to the legal issues with regard to smart contracts, electronically concluded contracts through the use of the internet were not regulated initially. The legislative framework was adapted accordingly, in response to these developments. Hence, there's a real need for further successful smart contract applications, proving their commercial value and, consequently, stimulating a more adequate regulatory intervention.

[238] *Supra,* 0.
[239] J.S. GANS, "The Fine Print in Smart Contracts", *supra* from 173, 3 January 2019, 22.

TITLE V. DRAFTING SMART CONTRACTS

The limits of my language mean the limits of my world

- Ludwig Wittgenstein's "Tractatus Logico Philosophicus," published in 1921

107. Title IV gave an overview of smart contract possibilities in a broader sense. It displayed the importance of the creation of new commercial applications. This title will embark on a developing area of expertise: the creation of smart contract clauses. Such expertise is indispensable with regard to the creation of new commercial applications of smart contracts.

The nature of smart contracts was already discussed in Title III.[240] Inevitably, such nature will collide with contract clauses that are drafted at present. Therefore, an assessment will be made of the reconcilability of clauses in natural language with clauses in code.

Firstly, as smart contracts are essentially computer programs, chapter one will provide a condensed insight into the logic behind computer programs and computer languages. We will evaluate which languages are suitable for the creation of smart contracts. Subsequently, chapter two will compare program language with natural language in terms of vagueness and ambiguity. Legal texts are usually full of ambiguous terms, such as "promptly". Computers do not respond well to any form of ambiguity. Furthermore, we will suggest solutions to quantify abstract legal concepts, such as reasonableness, in order to implement these concepts in smart contract clauses.

Finally, chapter three will integrate several commercial contract clauses into the contract model which was presented in Title IV. Several examples of pseudo-code will be given to demonstrate the logic behind program languages and how it can be incorporated in smart contracts. In doing so, we will be able to answer the central research question of this dissertation: whether or not contracts can be transformed into smart contracts.

[240] *Supra,* 40.

CHAPTER 1. PROGRAMMING LANGUAGE

108. Already in the 1840's, Ada LOVELACE wrote the very first computer program, as she recognised that the machine had more capacities than merely calculating variables. LOVELACE laid down the brass tacks of a new science that would revolutionise the world.[241] Currently, Ada's name lives on as an American programming language. Throughout the 20th century, major programming language such as C, Java and C++ were created. Just like Ada, plenty of computer languages are actively used for a variety of computer tasks and programs. Each language has its own specifications, advantages and disadvantages. However, there are some overarching principles which directly stem from their nature as a computer language. This chapter will dive into these fundamentals from the perspective of a legal practitioner. Key concepts will be explained and situated within a broader context.

1.1. Computer program

a. A set of instructions

109. A program in its generic form is a set of instructions designed to achieve a certain objective. A simple example of a program is giving directions to someone on the street.[242] If one of the directions is wrong, or a key instruction was left out, the program will not succeed in achieving its objective: arriving at the right place.[243] From a mathematician's point of view, a program is a function that externalizes a relationship between a set of allowable inputs and the related outputs, computed by the function.[244] A computer program or an algorithm is essentially a set of instructions, readable and executable by a computer, resulting in a specific task or the solution to a predetermined problem.[245] These instructions are directed towards the CPU, the central processing unit of a computer.[246] If anything goes wrong during the execution of the computer program, similar to giving a wrong direction in the example above, the program will fail to accomplish its task. This is called a "bug" in computer science terminology.[247]

---

[241] E.E. KIM en B.A. TOOLE, "Ada and the first computer", *Scientific American*, 1999, vol. 5, 76–81.
[242] F. GLASSBOROW, *You Can Do It!: A Beginner's Introduction to Computer Programming*, West Sussex, John Wiley & Sons, 2004, 1.
[243] *Ibid.*
[244] J. CONNOLLY en D. COOKE, "The pragmatics of programming languages", *Journal of the International Association for Semiotic Studies*, 2004, 152-153, https://www.degruyter.com/view/j/semi.2004.2004.issue-151/semi.2004.065/semi.2004.065.xml (Consulted on 8 April 2019).
[245] A.B. TUCKER, *Computer science handbook*, Florida, CRC press, 2004, 129; M.J. ROCHKIND, *Advanced UNIX programming*, *supra* from 82, 2004, 1.1.2; It should be noted that there is a slight semantic difference between a computer program and an algorithm. However, for the purpose of this research, these are considered one and the same.
[246] ISDA e.a., "Smart Contracts and Distributed Ledger – A Legal Perspective", *supra* from 66, 2017, 15.
[247] F. GLASSBOROW, *You Can Do It!: A Beginner's Introduction to Computer Programming*, *supra* from 230, 2004, 1.

Computer programs appear in many forms and shapes. They possess certain specific characteristics which should be kept in mind during the stage of creation. For example, an important aspect to take into account is the necessity of an anticipating approach. One should consider unexpected events and plan ahead how the program can respond to those events, for example a computer freeze.[248] Another feature of computer programs is the required application of computational thinking.[249] Due to the nature of programs, the solution to a problem or a desired outcome can be achieved by separating the problem or initial state into different components. Just like instructing someone how to put together a piece of furniture, computer programs are incapable of solving a big problem at once. Hence, specific instructions are addressed towards each individual component. A pragmatic approach on the part of the programmer, is therefore indispensable.

110. Programs essentially comprise two parts: Statements and expressions. These two notions will be discussed separately.[250]

b. Statements

111. Statements are snippets of code that make the computer perform an action.[251] CPU's are extremely intolerant towards ambiguity. Therefore, statements enable the CPU to clearly distinguish whether or not an action should be taken.[252] Statements do not return any value, but are standalone units of execution. The most common example of a statement is a conditional statement, such as an "IF-statement".[253] Such statement falls under the category of control flow statements, which refers to the order in which instructions are executed or evaluated.[254] An IF-statement places the program at a crossroads and determines whether to go right or left, according to whether the statement is true or false.[255, 256] Another example of statements are loops. Loops enable the consecutive execution of a piece of code, until certain conditions are met.[257]

---

[248] As pointed out by D. MALAN during his lecture "CS50 – Introductory lesson on Computer Science", Harvard University (online), 22 December 2018.
[249] *Ibid.*
[250] C. DANNEN, *Introducing Ethereum and Solidity: Foundations of Cryptocurrency and Blockchain Programming for Beginners*, *supra* from 84, 2017, 52.
[251] *Ibid,* 52.
[252] *Ibid.*
[253] B.W. KERNIGHAN en D.M. RITCHIE, *The C programming language second edition*, New Jersey, Prentice Hall, 2006, 55.
[254] C. DANNEN, *Introducing Ethereum and Solidity: Foundations of Cryptocurrency and Blockchain Programming for Beginners*, *supra* from 84, 2017, 3.
[255] G. WILLOUGHBY, *Purebasic: A Beginner's Guide to Computer Programming*, Fegersheim, Aardvark Global Publishing, 2006, 38.
[256] One example of the implementation of an IF-statement is the piece of code that executes consequently thereby exercises a purchase option if a certain price point is reached.
[257] C. DANNEN, *Introducing Ethereum and Solidity: Foundations of Cryptocurrency and Blockchain Programming for Beginners*, *supra* from 84, 2017, 3.

c. Expressions

112. An expression is anything that can be evaluated to one value.[258] Expressions can be distinguished from statements because expressions do not lead to any execution. They render a value, as opposed to a designated action. Examples of expressions in code are: “a + b” or more complicated expressions, such as “5 + square (9 - 3) + square(square(2))”. The latter expression can be resolved by subdividing the expression into multiple subexpressions. A particular type of expression is a Boolean expression. Similar to an IF-statement, a Boolean expression renders a value which can either be true or false.[259] Often, Boolean expressions are accompanied by operators, such as >, <, =, AND, OR and NOT. For instance, the evaluation of Boolean expression “5 < 3” will lead to a Boolean value which is false.

d. Functions

113. Functions are particular types of statements that are recognized by the computer and are automatically connected to a set of related statements, specific to the function. A real live example would be asking a blind man to open a door. The blind man is familiar with opening a door (i.e. he knows he should push the handle and subsequently push the door open). He only needs to know which door should be opened, and where the door is located, if someone would give him instructions.[260] Because the man is familiar with the action of opening a door, he will be able to perform the same task over and over again. In this regard, a function can be described as a shortcut, as less statements need to be used to complete the same task.[261] Some functions result in no end result, but only generate a building block for a following statement. An example of a computer program function is typing a character on a keyboard in a word processor. The computer’s software recognizes the input (typing a character), applies the function and subsequently displays a character on the screen (return value).
The integration of functions enables a more efficient and streamlined process of writing a program, and will therefore be important for smart contract implementation on a larger scale. Specific functions might be created to serve smart contract purposes.

---

[258] *Ibid*, 52.
[259] G. WILLOUGHBY, *Purebasic: A Beginner’s Guide to Computer Programming*, *supra* from 242, 2006, 37.
[260] F. GLASSBOROW, *You Can Do It!: A Beginner’s Introduction to Computer Programming*, *supra* from 230, 2004, 36.
[261] G. DOWEK, *Principles of programming languages*, France, Springer Science & Business Media, 2009, 20.

1.2. Computer language

a. *Types of code*

i. *Machine code – Binary*

114. Instructions directed to the CPU cannot be transferred in natural language, as the CPU requires exact instructions and is unable to comprehend words in human language. Furthermore, CPU's are only capable of understanding binary language.[262] Binary is a numeric system in which numbers can only contain two values: zero or one (0 or 1). A parallel can be drawn between flicking a switch on and off.[263] Moreover, computing memory is represented in a stacked form containing multiples of eight numbers, either zero or one.[264] Writing programs in machine code is thus nearly impossible, due to its unintelligible nature.[265] Therefore, more sophisticated systems were developed to enable a more functional programming approach.

ii. *Source code – Syntax and semantics*

115. Source code, being the flipside of machine code, is more familiar to human language. It refers to a piece of code that comprises the original building instructions of a program.[266] Source code is used by programmers to write programs from scratch, or adapt existing programs. Once source code is written down and ready to execute, it is sent to a compiler. The compiler is a software development tool that converts the source code into machine code, which then can be executed by the CPU.[267]
Statements, expressions and functions will be drafted in accordance with the syntax of the program language. The functions will be derived from a so-called library, which is introduced into the code and refers to the place where specific functions can be accessed.[268] Each program language has its own syntax, which refers to the rules that structure the format of instructions expressed in source code.[269] For example, the syntax of a language will determine whether or not parentheses are ought to be used when an IF-statement is expressed. Most languages maintain a syntax which is rather similar to natural language.

---

262 G. WILLOUGHBY, *Purebasic: A Beginner's Guide to Computer Programming*, Fegersheim, Aardvark Global Publishing, 2006, 262.
263 C. DANNEN, *Introducing Ethereum and Solidity: Foundations of Cryptocurrency and Blockchain Programming for Beginners*, *supra* from 84, 2017, 51.
264 For example, the binary translation of "Law" is 01001100 01100001 01110111.
265 ISDA e.a., "Smart Contracts and Distributed Ledger – A Legal Perspective", *supra* from 66, 2017, 15.
266 G. WILLOUGHBY, *Purebasic: A Beginner's Guide to Computer Programming*, *supra* from 242, 2006, 328.
267 F. GLASSBOROW, *You Can Do It!: A Beginner's Introduction to Computer Programming*, *supra* from 230, 2004, 8; G. WILLOUGHBY, *Purebasic: A Beginner's Guide to Computer Programming*, *supra* from 242, 2006, 320.
268 As pointed out by D. MALAN during his lecture "CS50 – Introductory lesson on Computer Science", Harvard University (online), 22 December 2018.
269 D.P. FRIEDMAN, M. WAND en C.T. HAYNES, *Essentials of programming languages*, Cambridge, MIT press, 2001, xvi.

Contrary to the syntax of a language, the semantics of a language determine the meaning of program elements.[270] Both syntax and semantics need to be examined thoroughly to create an executable program that enables achieving the intended tasks.

iii. Pseudo-code – Legible code

116. Pseudo-code is different from machine- and source code, as it cannot be deployed to effectively provide instructions to the CPU. Instead, pseudo-code can bring clarity to the thought pattern of a program language, ignoring the essential syntax related to a particular program language.[271] It is commonly used by programmers as a draft version of the code, being a mechanism to approach logical problems.[272] In chapter three, we will provide several examples of pseudo-code.

b. Types of languages

117. Throughout history, a significant amount of programming languages were developed, each containing their own syntax and semantics. The 20th century brought the arrival of the first high-level general purpose programming languages, such as Fortran.[273] These languages can be categorized in different types, such as educational languages[274], scripting languages[275] and curly-bracket languages.[276] As mentioned before, Solidity is the most common language with regard to the creation of smart contracts.[277] Usually, Solidity is used on the Ethereum blockchain, because the source code can be compiled into Ethereum Virtual Machine code, or EVM-code.[278] It is assumed to be a contract-oriented programming language, borrowing significantly from its predecessor, object oriented programming languages.[279] This implies several interface specifications which optimize the possibilities of programming contracts.[280] Solidity has substantial resemblances with JavaScript and C, which are some of the most used languages around the world.[281] Therefore, the program language is quite approachable for anyone with only a little experience in programming. Besides Ethereum, other platforms can be used

---

[270] *Ibid.*
[271] *Supra,* 115
[272] F. GLASSBOROW, *You Can Do It!: A Beginner's Introduction to Computer Programming*, *supra* from 235, 2004, 22.
[273] A.B. TUCKER, *Computer science handbook*, *supra* from 232, 2004, 18.
[274] An example of an educational language is Scratch. For more, see https://scratch.mit.edu.
[275] An example of a scripting language is PHP.
[276] An example of a curly-bracket language is C.
[277] *Supra*, 41.
[278] V. BUTERIN, "A next-generation smart contract and decentralized application platform", *supra* from 7, 2014, 17.
[279] QUALIUM, "Solidity – Ethereum's Programming Language for Smart Contraction", *Qualium Systems*, 18 July 2018, https://www.qualium-systems.com/blog/useful-it-articles-and-advices/solidity-ethereums-programming-language-for-smart-contraction/ (Consulted on 18 April 2019).
[280] *Ibid.*
[281] C. DANNEN, *Introducing Ethereum and Solidity: Foundations of Cryptocurrency and Blockchain Programming for Beginners*, *supra* from 84, 2017, 71.

to deploy smart contracts as well. For example, Hyperledger Fabric supports the languages Java and Go.[282]

CHAPTER 2. REMEDYING VAGUENESS

118. Before this research will analyse individual types of clauses, this chapter will give a more general overview on vagueness and ambiguity in contracts. Legislations are bursting with these vague terms that require further interpretation by parties and/or courts. As mentioned before, ambiguity should be avoided at all costs when machine code is fed into the CPU. First, we will define vagueness. A few examples of vague concepts in the law will be mentioned. Next, we will evaluate whether such vagueness can be eliminated. Finally, possibilities on turning abstract concepts (e.g. reasonableness) into quantifiable, computer-understandable concepts will be presented. These possibilities will then be used in chapter three, to integrate abstract clauses, such as impediment, in smart contracts.

2.1. Defining vagueness

119. Vagueness refers to the situation in which there are interpretations by users that are characteristically undecided about the truth or the falsity of such interpretation.[283] For example, the colour blue can be considered vague. Even though blue can be distinguished from a distinctly different colour, such as yellow, more related shades such as turquoise and lavender might be categorised by some people as blue, but not by others.[284] Issues with regard to vagueness arise when a continuum is confronted with terminology that has, or aspires to have, a bivalent logic (i.e. the capacity of assigning a true or false value to each proposition).[285] In a legal context, concepts such as reasonableness, equality and good faith do not amount to a value which is either true or false. The law does not provide any specific quantifiable instructions on the interpretation of these concepts. They usually are interpreted on a case-by-case basis.

2.2. Elimination of vagueness

120. From a logical perspective, it is unclear whether vagueness can be eliminated through clarifications and interpretation guidelines in the law or in contracts. The Frigaliment Importing Co. v. BNS International Sales Corp case in the U.S. is an excellent example to demonstrate such conundrum. [286] The case dealt with the sale of chickens. The buyer expected that 'chickens' meant young chickens,

---

[282] B. SINGHAL e.a., "How blockchain works", *supra* from 29, 2018, 20.
[283] J. WALDRON, "Vagueness in law and language: Some philosophical issues", *Cal L. Rev.*, 1994, 513, https://heinonline.org/HOL/Page?handle=hein.journals/calr82&div=24&g_sent=1&casa_token=&collection=journals&t=1556982368 (Consulted on 4 May 2019).
[284] *Ibid.*
[285] *Ibid,* 516.
[286] Frigaliment Importing Co. ,Ltd. ,v. BNS InternationalSalesCorp., 190F.Supp.116, 117 (S.D.N.Y. 1960).

suitable for boiling and frying. Such chickens had a higher market value than the other type of chickens, which were stewing chickens. Subsequently, the seller delivered the latter category of chickens, leading to a dispute. No specific definition was adopted in the sales contract. Both seller and buyer were assuming that 'chicken' would not be ambiguous. However, the opposite was true. This illustrates that even the most conventional and trivial terms can be ambiguous.

Parties can attempt to anticipate such ambiguity by adopting definitions and clarifications. It should be noted that it is nearly impossible to anticipate every possible interpretation. This problem is inherent to the nature of natural language.[287] The meaning of words ultimately depends on how those words are used by different people. An ideal language which manages to guarantee a uniform and consistent interpretation of terms has been researched by logicians.[288] Important to keep in mind is that specifying every possible notion is not the solution to the vagueness issues of laws and contracts. There is no assurance that a reduction in generality corresponds to a reduction in vagueness.[289] For example, one can determine more easily whether or not something is a 'human community' as opposed to a city, because the latter should be distinguished from villages or hamlets. Only if certain conditions are met, one can confirm that such group of people can be classified as a city. 'City' therefore needs additional certain criteria to be fulfilled.[290] In such example, the use of the term 'city' will only increase the vagueness of a phrase.

We can conclude that it is unlikely for a natural language contract to eliminate every possible form of vagueness. This conclusion is drawn based on the social interpretation of natural language: by default, there's no single meaning of a word or expression.

121. On the other hand, CPU's interpret the inserted machine code in a uniform and consistent manner, as they are completely resistant against any form of social interpretation. The meaning of an expression or a statement is a technical fact, rather than a socially contingent fact.[291] However, it can be argued that the semantics of a program language cannot be determined by the program itself.[292] It requires a human intervention to create such semantics, and as explained before, such human intervention is based on a social interpretation, rather than a technical interpretation.[293] The meaning of programs created by humans is defined in terms of *something else created by humans (i.e. the program language)*.[294] Therefore, it should be noted that when a program language is altered by its creators, the meaning of the programs written in such program language might be altered as well. Over time, the

---

[287] J. GRIMMELMAN, "All Smart Contracts Are Ambiguous", *supra* from 113, 2019, 10.
[288] J. WALDRON, "Vagueness in law and language: Some philosophical issues", *supra* from 301, 1994, 522.
[289] *Ibid.*
[290] *Ibid.*
[291] J. GRIMMELMAN, "All Smart Contracts Are Ambiguous", *supra* from 113, 2019, 10.
[292] *Supra,* 102.
[293] J. GRIMMELMAN, "All Smart Contracts Are Ambiguous", *supra* from 113, 2019, 11.
[294] *Ibid.,* 12.

meaning of certain functions might be adjusted or updated, leading to a risk of unforeseen execution of the program.

122. Nonetheless, interpretation issues are far less likely in the context of program languages. Program languages are usually constructed based on a consensus between programmers. They are distinguished from natural languages, which are also based on a social consensus with regard to the meaning of words and expressions, because the meaning of the syntax of program languages is hard coded and is not open for debate or interpretation. The same functions will result in the same execution (i.e. contain the same meaning) in different contexts and programs. As a consequence, programming languages can create a stable environment which guarantees a certain interpretation. In the context of legally binding agreements, such environment enhances legal certainty.

We can conclude that programming languages therefore are a better tool to combat vagueness in contracts. Well-informed parties might benefit from such a characteristic, as the execution of a smart contract will not be subject to interpretation issues.

2.3. Semantic issues [295]

123. We concluded that smart contracts can provide an advantage in harmonizing the interpretation of the parties' agreement. As mentioned before, in order to successfully execute a computer program, binary machine code should be fed into the CPU.[296] To benefit from the interpretation advantage of smart contracts, the adoption of the parties' agreement in smart contract form will need to anticipate the requirement of ultimately being conversable in binary language. In other words, there is no tolerance for a grey area: the elements which are agreed upon should be either black or white (i.e. either 0 or 1). This leads to several semantic issues that need to be resolved for an accurate and efficient adoption of an agreement in smart contract form.

a. Ambiguity

124. Contracts usually contain a substantial amount of potentially ambiguous terms. At first sight, these terms are not likely to cause any complications. For example, the use of "OR" in natural language can refer to the situation in which a certain action is undertaken when one premise or the other premise is true. However, it is questionable whether or not the action will be taken if both are true.[297] Computer languages might approach these issues differently. According to the general principles of logic, when

[295] C.D. CLACK, "Smart Contract Templates: Legal semantics and code validation", *supra* from 91, 2018, vol. 4, 346.
[296] *Supra*, 114.
[297] C.D. CLACK, "Smart Contract Templates: Legal semantics and code validation", *supra* from 91, 2018, vol. 4, 346.

both propositions are true, action will be taken anyway.[298] Another example are time-related expressions, such as "promptly" or "soon". A computer program cannot recognise parties' intentions that are expressed with such terminology. Hence, parties should be aware of these ambiguous semantics when drafting smart contracts.

b. Translating abstract and non-quantifiable concepts to code

125. A more difficult matter is translating concepts which obviously require further interpretation to result in an executable concept. In both legislative instruments and contracts, these abstract concepts play a major role in regulating the behaviour of humans. For example, the abstract concept of *bonus pater familias* is used to determine whether the actions of a person were negligent. Another example is the concept of "good faith", according to which contracts should be executed while showing due regard for the interests of the other party. What is due regard will need to be assessed on a case-by-case basis.[299] One should approach these concepts and find a solution to quantify their application. Several mechanisms can be distinguished which might enable the translation of these abstract concepts.

i. Quantification through machine learning

126. One way to approach the issue of translating abstract concepts is by using an algorithm which determines their quantification, based on machine learning technology. Machine learning was already explained in Title IV.[300] It involves computer algorithms that have the ability to learn and improve in performance over time.[301] The algorithm's ability to learn is based on data which is extracted from comparable cases. 'Learning' is rather a metaphor, as the computer system is only able to alter its behaviour and decision-making process through experience.[302] Machine learning algorithms could be used *a priori* and *a posteriori* the drafting of a smart contract. As mentioned before, parties could use the technology to generate recommendations concerning specific parameters of their agreement during the stage of negotiation.[303] For example, a machine learning algorithm could present a recommendation for a price, based on previous experiences with the sale of comparable goods.

The algorithm can also be used to clarify non-defined abstract concepts, such as 'reasonably' or 'promptly'. For example, if a certain contract does not specify a delivery period, a reasonable period might be determined by the machine learning algorithm.[304] Similar to a judge's intuition, the machine

---

[298] V. FACK, *Algoritmen en datastructuren*, Leuven, Acco, 2011, 6.
[299] C. VON BAR e.a., *Principles, definitions and model rules of European private law: draft common frame of reference (DCFR)*, *supra* from 18, 2009, 304.
[300] *Supra,* 58.
[301] H. SURDEN, "Machine learning and law", *Washington Law Review*, 2014, 87, https://heinonline.org/HOL/Page?handle=hein.journals/washlr89&div=7&g_sent=1&casa_token=UMliB43s3moAAAAA:0XDk0lWORC1KDrPF-kYYBRB8zrr-FN45Xh1Kdb5G8x_o62AlJt13IEzQ5yxJHPZNm3_OaZwy (Consulted on 18 April 2019), 88.
[302] *Ibid.,* 88.
[303] *Supra,* 77.
[304] *Infra,* 133.

learning algorithm develops a certain framework that serves as a point of reference for new, comparable cases. We will not analyse potential (human rights) conflicts that are caused by the machine replacing the court to make certain decisions. This falls beyond the scope of our research. We can simply conclude that, from a technical perspective, machine learning algorithms might be able to quantify abstract concepts and create a reference point to which a specific case can relate. One should take into account that this only applies to cases which show a sufficient degree of comparability. Chapter three of this Title will elaborate on the application of machine learning technology with regard to specific types of contract clauses.

ii. Quantification through oracle input

127. A second method to quantify legal abstract concepts is by using oracles that can provide an input in the smart contract. In doing so, the oracle's input replaces the abstract concept as if it were included in the smart contract *ab initio.* The oracle can take several forms. A first form is the data input by an expert on the subject. Such an expert could, for example, determine a reasonable price of a second-hand car. Evidently, the security and the effectiveness of a smart contract depends on the reliability of such an expert. One could question whether or not the implementation of an expert as an oracle opens the door to corruption in the smart contract setup. If one of the contracting parties can manipulate the expert by financially incentivising him, the data input will be corrupt. The reliability of such subjective oracle can be increased by appointing an expert panel.[305] Furthermore, the identity of the members of such panel can be pseudonymised. The panel can be paid based on their track record of earlier decisions. Also, in the context of dispute resolution, ad hoc arbitration panels might resolve potential conflicts. BUTERIN, the founder of Ethereum, refers to such arbitration panels as 'decentralised courts'.[306] For example, when one party claims that the counterparty has acted against the principle of good faith, such dispute could be brought before an online dispute resolution mechanism.

128. Apart from an expert, an expert panel or an ad hoc arbitration panel, one can also consider implementing a court as an oracle. In the event a dispute arises, the court's ruling can work as a data input in the blockchain ledger. The decision can affect the execution of the smart contract, or can activate certain parts of the code, such as a coded compensation clause. However, several concerns can be raised with regard to such a system. Relying on courts to provide certain decisions that affect the execution of a smart contract contradicts the initial purpose and philosophy of a decentralised network and the autonomous execution of smart contracts. As mentioned before, the essential value of smart

---

[305] A. WRIGHT en P. DE FILIPPI, "Decentralized blockchain technology and the rise of lex cryptographia", *SSRN*, 2015, 50, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2580664 (Consulted on 4 May 2019).
[306] I. KAMINSKA, "Decentralised courts and blockchains", *Financial Times*, 26 April 2016, https://ftalphaville.ft.com/2016/04/29/2160502/decentralised-courts-and-blockchains/ (Consulted on 4 May 2019).

contracts resides in its ability to execute itself, without the need for recourse to a court. A great deal of value is lost when such automated execution can be affected by the court's decision. Also, at present, courts aren't adapted sufficiently to deal with smart contracts. On the one hand, they aren't technically equipped to support such a setup. They should be able to integrate rulings into a digital, decentralised environment to fulfil their role as smart contract oracle. On the other hand, legal systems require certain formalities to be held, in order to bring a case before a court. For example, the conventional way to bring a case before a court is by means of a summons. It is currently impossible to automate such a summons.

CHAPTER 3. MAPPING CONTRACT CLAUSES

129. This chapter will analyse, discuss and evaluate contract clauses which often form the backbone of commercial contracts. Such analysis will be done in three steps. Firstly, the type of contract clause will be explained from a legal perspective and will be clarified by examples in legal prose. As the scope of this research does not allow for an exhaustive analysis of all possible contract clauses, the extent will be limited to those clauses which might be important for further smart contract development. No particular legal system will be used as a reference point. Instead, the Draft Common Frame of Reference (hereafter: "DCFR"), the Principles of European Contract Law (hereafter: "PECL") and the United Nations Convention on Contracts for the International Sale of Goods (hereafter: CISG) will be used as reference points in order to draw conclusions from a legal perspective. [307]

Secondly, an evaluation will be made of the intrinsic ability of those clauses to be transformed in coded clauses. Where possible, examples will be given in pseudo-code, to enhance legibility and to avoid concentrating on one particular program language.[308]

Thirdly, the clause will be integrated into the contract model, as described above, in accordance with the results from the analysis in step one and two.[309]

3.1. Time related clauses

a. Legal significance

i. Term clauses

130. A term clause in the contract indicates the period during which the contract will render legal rights and obligations for the contracting parties. An example is the period during which a license can

[307] C. VON BAR e.a., *Principles, definitions and model rules of European private law: draft common frame of reference (DCFR)*, *supra* from 18, 2009; O. LANDO e.a., *Principles of European contract law: Parts I and II*, *supra* from 18, 2000; United Nations Convention of 11 April 1980 on Contracts for the International Sale of Goods.
[308] *Supra*, 116; V. FACK, *Algoritmen en datastructuren*, Leuven, Acco, 2011, 6-7.
[309] *Supra,* 75.

be used by the licensee.[310] Contracts have either a fixed or a non-fixed term. The latter category is subjected to a universally adopted principle, which stipulates that such contracts can be terminated immediately by one of the contracting parties.[311] Term clauses are of particular relevance for non-expiring agreements.[312] An example of an expiring agreement is a conventional purchase agreement which is executed immediately. The contract is terminated when the buyer pays the price, and the seller delivers the good. Such contracts come about on a daily basis. Non-expiring agreements (or continuing agreements), such as the lease of an apartment, have a longer period of execution. In such case, it is relevant to specify whether or not the agreement has a fixed term, as this determines whether or not one of the parties can unilaterally terminate the contract.[313]

ii. Time of performance

131. Parties can agree on the moment during which the contract will be executed. Also, as opposed to indicating a particular moment, the agreement can specify during which timeframe certain obligations should be performed by one of the parties. For instance, the online purchase of a good usually initiates a predetermined delivery period during which the seller can perform his obligation to deliver the good. Finally, if no particular moment in time or period is determined, performance is to be made within a reasonable time after the obligation arises.[314] A reasonable time is interpreted according to the circumstances of the case.[315]

b. Coding possibilities

132. Time related clauses rely on an element which cannot be manipulated by either party. No recourse to trusted third parties is required, as time and date are elements which are backed by a universal consensus. Furthermore, time is an intrinsic part of the blocks in a blockchain.[316] Therefore,

---

[310] *Supra*, 86.
[311] Cass. (BE) 22 November 1973, *Pas.* 1974, I, 312; R. DEKKERS, 'De la rupture des contrats à dureé illimiteé', *RCJB* 1957, 316; H. DE PAGE, o.c., II, nr. 763; X. DIEUX, 'Observations sur l'article 1794 du Code civil et sur son champ d'application', *RCJB* 1981, 528.
[312] M. DE POTTER DE TEN BROECK, R. DE CORTE, B. DE GROOTE en D. BRULOOT, "Verbintenis uit overeenkomst" in *Privaatrecht in hoofdlijnen*, Antwerpen, Intersentia, 2017, 235.
[313] Fixed term contracts are subject to the adage "pacta sunt servanda". The refers to the impossibility of unilaterally terminating the contract before its term expires, unless such possibility is incorporated in the contract, or the law allows to do so. See C. VON BAR e.a., *Principles, definitions and model rules of European private law: draft common frame of reference (DCFR)*, *supra* from 18, 2009, 319. ("III.–1:109: Variation or termination by notice: (1) A right, obligation or contractual relationship may be varied or terminated by notice by either party where this is provided for by the terms regulating it. (2) Where, in a case involving continuous or periodic performance of a contractual obligation, the terms of the contract do not say when the contractual relationship is to end or say that it will never end, it may be terminated by either party by giving a reasonable period of notice.")
[314] C. VON BAR e.a., *Principles, definitions and model rules of European private law: draft common frame of reference (DCFR)*, *supra* from 18, 2009, 331. ("III.–2:102: Time of performance (1) If the time at which, or a period of time within which, an obligation is to be performed cannot otherwise be determined from the terms regulating the obligation it must be performed within a reasonable time after it arises.")
[315] *Ibid,* 332.
[316] *Supra*, 14

we can conclude that it is possible to express time related clauses in Boolean logic.[317] A certain moment in time or a determined period in time can be evaluated to a value which is either true or false. This can be realized through the use of IF-statements and loops. In doing so, a certain part of the code will only be executed once the Boolean expression evaluates to true, implying that the predetermined moment or period arrives.

Algorithm 1: Contract term
Procedure: IsContractTermActive (StartDate, Duration, CurrentDate)
IF CurrentDate < StartDate + Duration
RETURN: TRUE

An example might be the exercise of an option during a predetermined timeframe. Whenever the stock price (p) of a share reaches a certain trigger point (t) during that particular period of time, the option will be exercised by the IF-statement in the smart contract.[318] If the expiration date of the option has passed, the option will no longer be exercisable.[319]

Algorithm 2: Exercise option
Procedure: ExerciseOption (Option, StartDate, Duration, CurrentDate, p, t)
IF CurrentDate < StartDate + Duration = TRUE then
IF p ≤ t
Exercise(Option)

Moreover, in case of a non-fixed term contract, both parties are entitled to terminate the contract. Such decision can amount to an effective smart contract clause as well. Both parties can be granted a termination trigger which interrupts the loop in the code and prevents it from further executing. This can also be constructed with a WHILE-statement, which is interrupted once one of the parties uses such trigger, or an IF-statement which can move the code into a different direction, as if it were a turn off switch. For instance, the latter case might be useful to determine certain consequences to the withdrawal of a license, originally given through a smart contract.

[317] *Supra*, 112.
[318] For the sake of clarity, it should be noted that the holder of an option has the right to exercise the option. This cannot be unilaterally enforced by the issuer of the option.
[319] H. SURDEN, "Computable Contracts", *supra* from 78, 2012, 659.

Algorithm 3: Contract termination
Procedure: TerminateContract (Contract, Boolean PartyA, Boolean PartyB)
1: IF PartyACancels OR PartyBCancels THEN
2: Terminate(Contract)

Hence, to a great extent, time related clauses contain the capacity of being casted in coded form.

133. Nonetheless, aforementioned conclusion should be nuanced. Certain aspects of time related clauses might be more problematic to adopt in smart contracts. Any construction that doesn't evaluate to a Boolean expression is subject to further scrutiny. Computational thinking strategies can be applied to divide a problem into multiple components that do translate to code.[320] One particular issue is the assessment of a reasonable term. The concept of "reasonableness" does not translate to binary, as it is simply not a matter of black or white. It implicitly refers to a fictional similar situation. This fiction includes the actions of a person in good faith, and heavily relies on the circumstances and the context of the case.[321] In other words, two aspects need to be taken into account to assess reasonableness: (i) the circumstances of the case and (ii) the behaviour of a person acting in good faith in these particular circumstances. These elements are rather vague with regard to providing accurate instructions to a CPU. Therefore, one should attempt to resolve this conflict by quantifying reasonableness in a way the CPU can grasp onto its boundaries and make an actual data-driven balancing exercise. In doing so, the grey area of reasonable behaviour can be categorized as either black or white, dependent on the case.
Such a result might be achieved through the use of machine learning technology.[322] In the context of time related clauses, machine learning technology might bring salvation to aforementioned conundrum by creating tangible boundaries. The algorithm itself can determine whether or not a performance is executed within a reasonable period of time, based on previous comparable cases.[323] Quantitative legal prediction, such as the determination of a reasonable period, has been used increasingly by lawyers and law firms.[324]

---

[320] *Supra*, 109.
[321] C. VON BAR e.a., *Principles, definitions and model rules of European private law: draft common frame of reference (DCFR)*, *supra* from 18, 2009; O. LANDO e.a., *Principles of European contract law: Parts I and II*, *supra* from 290, 2000, 3. (Article 1:302: "Under these Principles reasonableness is to be judged by what persons acting in good faith and in the same situation as the parties would consider to be reasonable").
[322] *Supra*, 57.
[323] For an interesting example of a statistical analysis on the reasonable notice period for the termination of distribution agreements, see C. VAN DER ELST en I. VANDEVELDE, "De redelijke opzeggingstermijn bij het beëindigen van de verkoopconcessie van onbepaalde duur: een objectieve concessieformule", *TPR*, 2014, vol. 4, 2487–2562.
[324] D.M. KATZ, "Quantitative legal prediction-or-how i learned to stop worrying and start preparing for the data-driven future of the legal services industry", *Emory LJ*, 2012, 912, https://heinonline.org/HOL/Page?handle=hein.journals/emlj62&div=27&g_sent=1&casa_token=S0xIbJI70RU AAAAA:0JUJMpSGeZ1P2kYOxuwtaQFu6wT591t6t5RYivgjG3SNTY7laEn6t_L3TyK_CP7Hnvunb7nt (Consulted on 18 April 2019).

Following algorithm presents an example of such reasonable period. When a certain threshold is reached, the algorithm will declare the time of performance (t) past a reasonable period, predetermined by a machine learning algorithm (r). As a consequence, a price reduction (p) will be applied and a warning will be sent to the seller.

Algorithm 4: Reasonable period

Procedure: ExecutionContract (Contract, StartDate (s) ReasonablePeriod (r), DateOfPerformance (d), InitialPrice (i), PriceReduction (p))

```
IF d < s + r
        PaySeller (i)
ELSE
        NotifySeller ( )
        ApplyPriceReduction (i – p)
```

134. Nonetheless, both legal and technical concerns might be raised. One should be aware of the difference in perspectives between a judge assessing the reasonableness from an ethical and philosophical perspective on the one hand, and an engineer assessing the reasonableness from a pragmatic and case study based perspective on the other hand.[325] Courts, having dealt with many different interpretations of the notion "reasonable", apply their own type of 'learning algorithm'. However, they also apply a social and psychological framework that considers circumstances which would slip through the net of a programmed machine learning algorithm. There's an important ethical decision to be made whether or not aforementioned threshold should be set by the engineer designing the machine learning algorithm. Another issue at hand is the prerequisite of similar circumstances. The machine learning algorithm is incapable of determining the difference between the home delivery of a pizza and the home delivery of a custom made bicycle. In other words, the machine learning solution will only work for those contracts which carry out a rather conventional and comparable obligation. For non-conventional contracts, the machine learning algorithm will not be able to determine an accurate threshold.

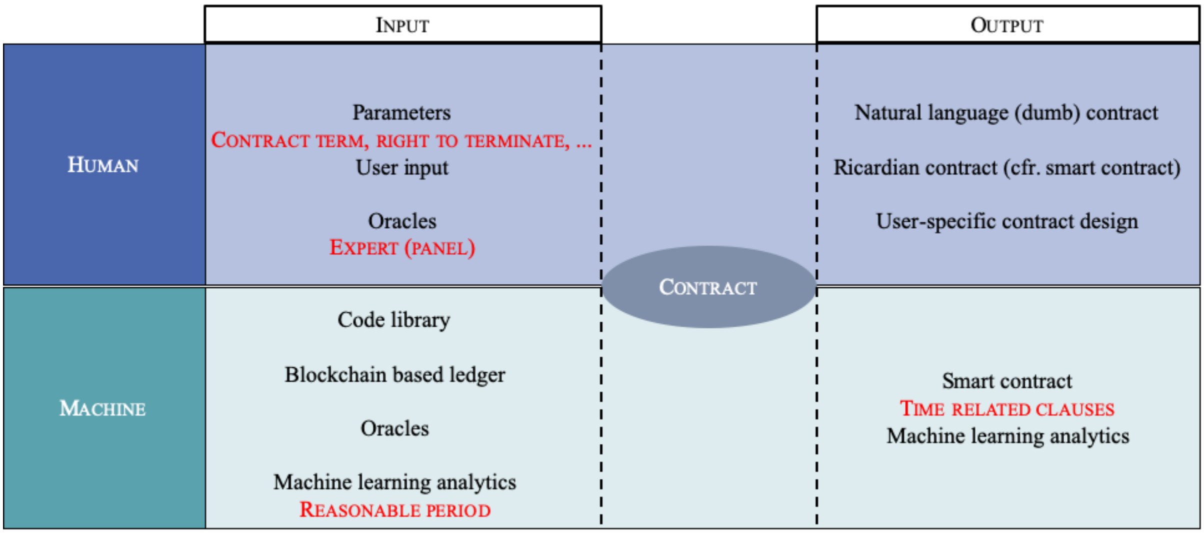
c. Integration in the new contract model

135. As mentioned before, a great deal of time related clauses can be embedded in the smart contract component of the model.[326] Specific functions can be created which are linked to user parameters, as

---

[325] W. WALLACH en C. ALLEN, *Moral machines: Teaching robots right from wrong*, New York, Oxford University Press, 2008, 75.
[326] *Supra,* 75.

made clear by the examples above. These parameters deal with time-related aspects such as the duration of the contract and the right to unilaterally terminate the contract. These elements are integrated into the smart contract by relying on the templates which are stored in the code library. For example, a licensor and a licensee can agree on the issuance of a license for the period of two years. The parameter "two years" will be linked to the corresponding function in the code library. Such construction facilitates a low entry level for parties, as they only have to indicate specific elements of the smart contract.



136. On the contrary, the parameters might determine when the intended agreement does not qualify for smart contract uses. For non-conventional contracts that do not indicate a fixed contract term, it might be opportune to compose a natural language (dumb) contract.

3.2. Financial clauses

a. Legal significance

137. Financial clauses refer to the monetary rights and obligations which are agreed upon by the contracting parties. They form a crucial part of commercial and financial contracts. As mentioned before, financial incentives were the engine that initiated the adoption of blockchains and, subsequently, smart contracts. As a consequence, an increasing amount of potential applications reside in the financial sector. This is mainly caused by the non-physical nature of money involved in many financial transactions. The existence of digital money depends on the ledger (i.e. bank accounts) in which it is registered. A substantial amount of centralised monetary transaction systems are already in place, such as mobile banking applications and contactless payment systems. This illustrates that the infrastructure and the technology to support automated financial transactions are well-integrated.

i. Payment clauses

138. Payment clauses refer to the performance of paying a price by one or more parties, and the relating manner, method and modalities of such a payment. Essentially, parties are free to agree on any price they desire, as the governing contract laws do not intent to interfere with market forces. However, specific rules affect the modalities of such payments. For example, a lot of smart contract transactions will involve the payment of a price in cryptocurrency value. The question might arise whether this is a valid method of payment. In this regard, the DCFR stipulates that "payment of money due may be made by any method used in the ordinary course of business".[327] For now, it's unclear whether cryptocurrency payments can be considered a method used in the ordinary course of business. However, the DCFR also states that "The development of new techniques for payment must not be prevented by a detailed enumeration of possible manners of payment".[328] One can argue that the proven scalability of Bitcoin and its fast transaction speed establish a valid method of payment. Other cryptocurrencies are therefore subject to the same reasoning. Furthermore, some businesses already accept cryptocurrency payments.

ii. Determination of price

139. In conventional circumstances, the parties that conclude the contract will determine and specify the price of a good or service in the contract. However, in some contracts, the price determination isn't specified in the terms of the contract, but refers to a price determination mechanism instead. According to the principles of European contract law, when a price has yet to be determined, parties are treated as having agreed on a reasonable price.[329] Also, in certain situations, a party is awarded the power to unilaterally determine the price. The same rule of reasonability applies to such construction: the party cannot determine a grossly unreasonable price.[330] This should also be taken into account if parties agreed on the determination of the price by a third party.[331] For example, parties could allocate the power to determine the price to an expert. In a digital environment, parties could even agree on a price

---

[327] C. VON BAR e.a., *Principles, definitions and model rules of European private law: draft common frame of reference (DCFR)*, *supra* from 18, 2009, 340. (III.–2:108: Method of payment); The PECL stipulates the same in article Article 7:107, 1. For more, see O. LANDO e.a., *Principles of European contract law: Parts I and II*, *supra* from 290, 2000, 23.

[328] C. VON BAR e.a., *Principles, definitions and model rules of European private law: draft common frame of reference (DCFR)*, *supra* from 18, 2009.

[329] O. LANDO e.a., *Principles of European contract law: Parts I and II*, *supra* from 18, 2000, 20. (Article 6:104 - Determination of Price: Where the contract does not fix the price or the method of determining it, the parties are to be treated as having agreed on a reasonable price.)

[330] C. VON BAR e.a., *Principles, definitions and model rules of European private law: draft common frame of reference (DCFR)*, *supra* from 18, 2009, 262. (II.–9:105: Unilateral determination by a party: Where the price or any other contractual term is to be determined by one party and that party's determination is grossly unreasonable then, notwithstanding any provision in the contract to the contrary, a reasonable price or other term is substituted.)

[331] *Ibid.* (II.–9:106: Determination by a third person (1) Where a third person is to determine the price or any other contractual term and cannot or will not do so, a court may, unless this is inconsistent with the terms of the contract, appoint another person to determine it. (2) If a price or other term determined by a third person is grossly unreasonable, a reasonable price or term is substituted.)

determination mechanism which relies on data input from an external source. For example, in the context of insurance, the price of an insurance policy could vary depending on the number of policyholders. Hence, the freedom of contract allows parties to set up any price determination mechanism they prefer. However, the price will be subject to scrutiny, whether or not a grossly unreasonable price is determined.

140. Specific to European Union legislation, the capacity of the contracting parties might affect the applicable laws. In particular consumer rights legislation can impose restrictions on the freedom of contract. In this regard, the directive on unfair terms in consumer contracts stipulates the prohibition for a seller to determine the price at the time of delivery or to unilaterally increase the price, without granting the right to the consumer to cancel the contract if the price is too high in relation to the initial price.[332] Hence, one should be aware of the potential capacity as a consumer when assessing the legality of price determination clauses.

iii. Escrow

141. In Title III, we already provided an example of escrow services in a smart contract form.[333] Such mechanism establishes a relationship of trust between a buyer and a seller. A trusted third party is responsible for releasing the purchase price once the seller has performed his obligation to deliver the good. Therefore, an escrow clause is considered a modality of the payment of a price. As mentioned before, escrow services are excellent opportunities to be adopted in smart contract form.

b. Coding possibilities

142. In general, financial clauses are to a large extent translatable to coded clauses, as the majority of financial assets are already virtually embedded in centralised systems. For instance, mobile banking apps allow users to automatically transfer money from one account to another. In the realm of decentralised ledgers, cryptocurrency transactions are proven to be successfully programmable, given the success of Bitcoin and Ether. Usually, cryptocurrency transactions are executed with the available funds in a user's wallet. Such a digital wallet is an electronic device which allows an individual to make electronic transactions.[334] A certain financial transaction will be executed if the payer's wallet contains sufficient funds. When deploying smart contracts on Ethereum, one should also take into account the amount of gas which should be paid to cover the cost of the transaction (i.e. the cost of running the smart contract on the blockchain).[335] This minor transaction fee belongs to the miner who successfully

---

[332] Article 3 (3) j° annex 1 (l) of Directive 93/13/EEC of 5 April 1993 on unfair terms in consumer contracts.
[333] *Supra,* 84.
[334] M. GIANCASPRO, "Is a 'smart contract' really a smart idea? Insights from a legal perspective", *supra* from 10, 2017, vol. 6, 828.
[335] *Supra,* 44.

solves the proof of work.[336] The following example displays a straightforward payment from buyer to seller, including the gas price which is borne by the buyer.

Algorithm 5: Payment I

```
Procedure: PayPrice (Contract, Price (p), Gas (g), WalletBalance (w) )
IF p + g < w
        RETURN: INSUFFICIENT FUNDS
ELSE
        PaySeller (w – p – g)
        NotifySeller ( )
```

143. One of the key selling points of blockchain based smart contracts is their ability to rely on oracles to determine when contract execution should or shouldn't take place.[337] Furthermore, such oracles can affect price determination mechanisms. For example, the exercise price of options belonging to an employee can be adjusted to the remaining period during which the employee is employed by the company. In this case, the oracle could be the National Social Security Service, as they are notified once the employee's employment contract is terminated. Another example is the price determination by an independent expert. In a situation where buyer and seller cannot agree on a price for a second hand car, a car mechanic can act as an oracle (i.e. an independent expert) to determine a reasonable price. However, as mentioned before, the determination of a price is restricted to a certain extent. The price cannot be grossly unreasonable.[338] The CPU cannot cope with the ambiguity related to such abstract form of a price limit. Therefore, the abstraction of 'unreasonableness' should be translated to a more tangible concept. Similar to the machine learning algorithms applied to determine a reasonable period for time related clauses, machine learning algorithms can also be used to determine a grossly unreasonable price. Evidently, the autonomy of parties should be respected. Hence, such algorithm can only be applied to prices which are yet to be determined with the conclusion of the contract. The machine learning algorithm will set a certain limit, indicating a reasonable price. However, the algorithm can only determine an accurate and fair limit to such price setting when the circumstances of the transaction are sufficiently comparable. Moreover, the algorithm requires a significant amount of comparable cases to determine such price limit.

---

[336] *Supra,* 18; C. DANNEN, *Introducing Ethereum and Solidity: Foundations of Cryptocurrency and Blockchain Programming for Beginners*, *supra* from 100, 2017, 43.
[337] *Supra,* 47.
[338] *Supra,* 139.

144. Machine learning algorithms can serve as an oracle by providing data input into the blockchain. Besides the application of a machine learning algorithm, other means can be devoted to solve the unreasonable price issue and give data input as an oracle as well. For example, an expert on the particular matter can act as an oracle to provide data input in the smart contract. Such mechanism might be useful in the situation where a price is determined by one of the parties after the conclusion of the contract, and the counterparty claims the price is grossly unreasonable.[339] Hence, the buyer is entitled to invoke the assessment of the price by an expert, which is predetermined by the smart contract. Upon the activation of such right, the expert will conclude whether or not the determined price is grossly unreasonable. As mentioned before, the reliability of such expert is crucial to maintain the integrity of the smart contract setup. The following snippet of pseudo-code demonstrates such mechanism:

Algorithm 6: Payment II

```
Procedure: PayPrice (Contract, Price (p), Gas (g), WalletBalance (w), Boolean Expert )
IF p + g < w
    RETURN: INSUFFICIENT FUNDS
ELSE
    IF
        ExpertApproves (p)
            PaySeller (w – p – g)
    ELSE
        RETURN: UNREASONABLE PRICE
```

145. Another possibility is a more delicate solution to the reasonableness issue. Similar to the insertion of the input of an independent expert in the network, parties could opt to have recourse to a court instead. The court acts as a thrusted third party oracle which can declare a certain determined price unreasonable. However, such a setup raises several concerns, as explained before.[340]

146. Smart contracts could include an escrow mechanism as a payment modality. Such escrow service can be rather easily translated to code. The mechanism of raising stakes to incentivise parties was already described in Title IV.[341]

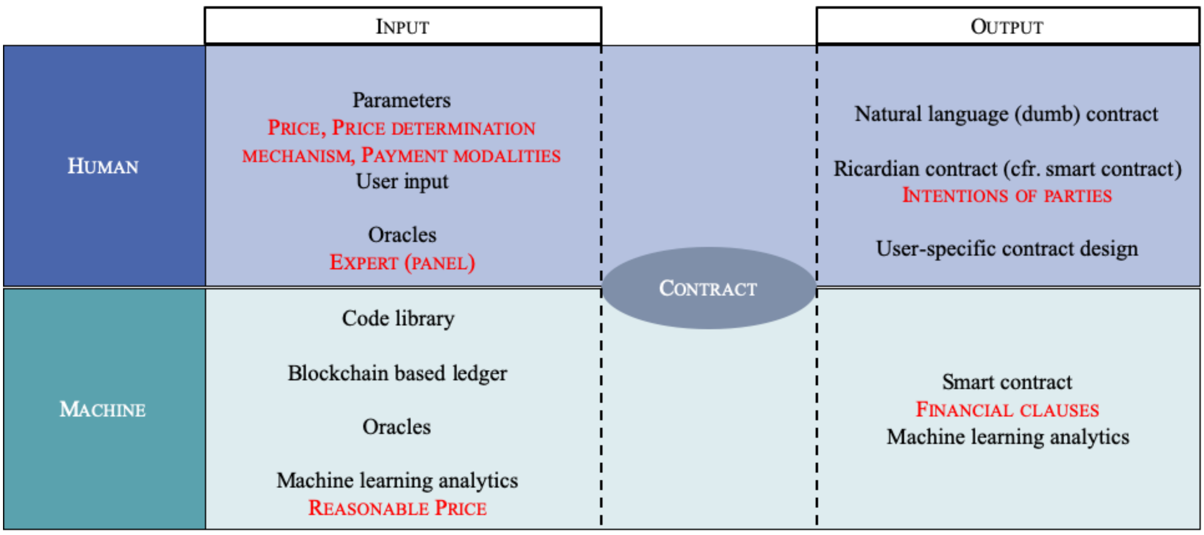
---

[339] A. WRIGHT en P. DE FILIPPI, "Decentralized blockchain technology and the rise of lex cryptographia", *SSRN*, 2015, 50, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2580664 (Consulted on 4 May 2019).
[340] *Supra,* 128.
[341] *Supra,* 84.

```
Algorithm 7: Escrow
Procedure: PayPrice (Contract, Price (p), Escrow )
EscrowSetup ( )
        BuyerInsertsStake (2 * p)
        SellerInsertsStake(2 * p)
        IF
                BuyerConfirmsDelivery
                        RETURN: DELIVERY CONFIRMED
                        PayStakeBuyer (1 * p)
                        PayStakeSeller (3 * p)
```

c. Integration in the new contract model

147. We conclude that, to a large extent, financial obligations are by nature translatable to smart contract clauses. Any conventional payment clause with a predetermined price can qualify as an operational clause, subject to the Boolean logic of carrying a value of either true or false. Therefore, these clauses will be mapped in the smart contract part of the model, as presented in Title IV. Users can adjust certain parameters in the input section, thereby providing instructions on the price and the payment modalities, such as an escrow or a specific price determining mechanism. The financial smart contract clauses will be accompanied by a Ricardian contract, externalising the intentions of the parties based on aforementioned parameters. With regard to the determination of the price, a certain form of ambiguity might arise when parties decide not to determine a price at the moment the agreement is concluded. In such event, the price should be reasonable, according to international commercial practice (i.e. when an instrument as the CISG is declared applicable to the contract). Reasonableness can be assessed by using a machine learning algorithm, if the smart contract is sufficiently comparable to previous contracts. If not, other solutions should be found to take such requirement into account. Expert panels, functioning as oracles, can be consulted to resolve the issue of reasonableness as well.



3.3. Clauses related to physical objects

148. Smart contracts reside in a digital environment which is detached from the physical world. A certain synchronisation between those two environments is required when smart contracts imply consequences to and depend on both virtual and physical aspects. As a result, a certain quantification needs to be attached to these physical aspects. For example, when a certain service is provided and subsequently, a payment is due for such service, there has to be a way of communicating with the smart contract that the payment should be executed. In Title IV, we already gave the example of supply chain networks that use blockchain based smart contracts to optimize the transfer of goods between multiple parties.[342] The physical transfer of such goods needs to be quantified in order to provide a data input into the smart contract. In other words, there should be a virtual representation of the physical goods. Both the original and the representation should stay synchronised to maintain the integrity of the network.

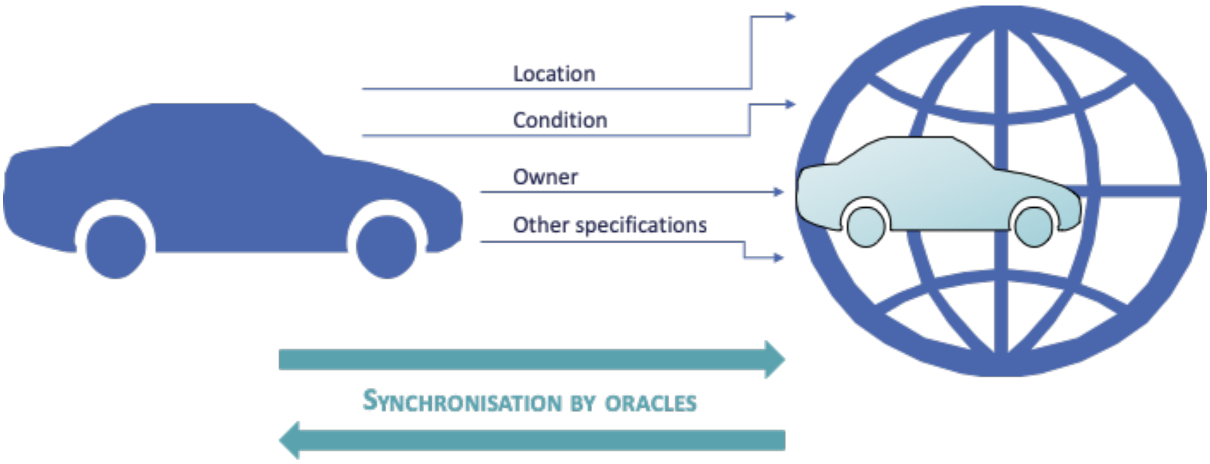


Figure 3 – Creating a virtual representation. In order to attach legal consequences in a smart contract to existing physical objects, a virtual representation on the blockchain network is required. Such representation is created by various specifications of the physical good, which are synchronized during the entire transaction by the use of oracles.

This paragraph will attempt to analyse the ability to code clauses which refer to physical objects. Among others, such category includes the transportation of goods, the construction or modification of goods, and the design of goods. The emphasis will be on those agreements which are presumed to be better candidates in terms of efficiency for a smart contract setup.

a. Legal significance

149. In a business to business relationship (hereafter: B2B), parties can determine a substantial amount of clauses without any restrictions, as there are few mandatory provisions governing their relationship. For the purpose of this research, we will consider the DCFR and the CISG as guidelines.

---

[342] *Supra,* 92.

Several legal provisions can play a role for agreements which are related to physical obligations to deliver certain goods. The seller should deliver the goods on the date which is adopted in the contract, or in case the contract lacks such date, within a reasonable period.[343] Also relevant is the passing of the risk on destruction of the goods when transporting the goods from one party to another. The DCFR states that the risk passes when the buyer takes over the goods or the documents representing them.[344] In international commercial contracts, parties might also opt to incorporate so-called “incoterms”, which determine specific tasks, costs and risks associated with the transaction.[345] Incoterms are expressed in three letter words, such as EXW and CPT. Each type represents different legal implications for a transaction.

As a consequence, once the risk has passed, the buyer is obliged to pay the price.[346] Only one exception is made: when the loss or the damage is caused by the seller. Another rule to take into account is the buyer’s right to notify the lack of conformity of the goods. The buyer should examine the goods within as short a period as is reasonable in the circumstances.[347] In any case, such a period is limited to two years from the date on which the goods were handed over.[348] A lack of conformity that exists at the time of the passing of the risk results in the liability on the account of the seller.[349]

As mentioned before, B2B transactions are subject to a more lenient legal framework as compared to business to consumer transactions (hereafter: B2C). European legislation provides several mandatory provisions on the sale of goods to consumers in the Directive 2011/83/EU of the European Parliament and of the Council of 25 October 2011 on consumer rights (hereafter: Consumer Rights Directive). Especially in the context of distance contracts, consumers are entitled to certain rights which need to be implemented in smart contracts as well. The most important provision is the ability for consumers to withdraw from a distance contract within a period of 14 days.[350] Such withdrawal can succeed without giving any reason and implies the reimbursement of the consumer. The consumer is liable for any diminished value of the goods, except for the handling that is necessary to establish the nature, the characteristics and the functioning of the goods.[351]

---

[343] Article 33 CISG.
[344] C. VON BAR e.a., *Principles, definitions and model rules of European private law: draft common frame of reference (DCFR)*, *supra* from 18, 2009, 626.
[345] For more, see J. RAMBERG, “INCOTERMS 2010”, *Eur. JL Reform*, 2011, 380, https://heinonline.org/HOL/Page?handle=hein.journals/ejlr13&div=29&g_sent=1&casa_token=ozxVMlXCSJ0AAAAA:glq4ksO9vKgrVFAPz_AbgCiTpRR-OH_phXWbbqQyan0AiV0CCb3vv1WWQOX-E8Qom8vo_yrZ2A&collection=journals&t=1557489335 (Consulted on 10 May 2019).
[346] *Ibid.,* 624.
[347] *Ibid.,* 616.
[348] Article 39 CISG.
[349] Article 36 CISG.
[350] Article 9 Consumer Rights Directive.
[351] Article 14.2 Consumer Rights Directive.

150. Apart from the physical obligation to deliver the goods, and the related legal consequences, several other tangible aspects might be connected to smart contracts as well. For example in insurance contracts, the relevant insurance payment might depend on the realisation of a physical event, such as a car crash or a theft.

b. Coding possibilities

151. Evidently, things get more complicated with regard to the implementation of real world activities in smart contracts. Different types of oracles will be crucial to underpin such transactions, as they bridge the gap between the virtual and the physical world. The technical limitations and the reliability of oracles are therefore determining whether or not a certain smart contract setup might work, keeping in mind the legal requirements which are imposed on specific types of contracts. Both machine-based oracles, such as sensors, and human-based oracles, such as trusted third parties, should be assessed how they can interact with the virtual environment of smart contracts. The former category refers to any data which is generated by machines registering specifications of the agreement. For example, in an insurance context, the damaged car might have installed several sensors that can indicate the damages and the financial costs related to such damages. This data can be fed into a smart contract between the policy holder and the insurance company to release the insurance payment.
Machine-based oracles can only be effective if two requirements are met: the oracle should be accurate, and the oracle should be reliable. As mentioned before, at all times, parties should attempt to synchronize the physical, real world version of an object or objects with the virtual representation of the object or objects, represented in the smart contract. Only when such representation is sufficiently accurate and reliable, legal consequences can be attached to it, such as the passing of the risk of the goods.

i. A sufficient level of accuracy

152. Firstly, the oracle should satisfy a sufficient level of accuracy in determining whether or not certain events occurred. This should be evaluated on a case-by-case basis. For example, when buyer and seller agree that the risk passes upon the arrival of a cargo ship, it is fairly easy to implement a detection mechanism that registers the ship once it enters the docks, thereby passing the risk of the goods to the buyer. Another example is the tracking of temperature sensitive goods. This could be an interesting price determining modification for the supply chain industry, especially for food and medical transportation. Buyer and seller can agree to transport the goods in certain circumstances to safeguard their quality, for example by agreeing on a certain temperature at which the goods should be transported. A significant deviation of such temperature could come with a price reduction. A temperature sensor, integrated in the transportation carriage or container, can ensure such mechanism.[352] In the algorithm

---

[352] K. WÜST e.a., "Do you need a Blockchain?", *supra* from 55, 2018, 49.

below, a higher price of 10 % per degree Celsius is billed per hour when the transportation temperature deviates from the temperature in the agreement.

Algorithm 8: Temperature

Procedure: DeterminePrice (Contract, PricePerHour (p), Temperature (T), TemperatureMax (Tmax), TemperatureMin (Tmin), DurationInHours (h))

```
TransportProcedure ( )
        IF
                Tmin ≤ T ≤ Tmax
                Bill (p * h)
        ELSE
                IF
                        T < Tmin
                        Bill ((|(T-Tmin)|/10)+1) * p * h
                IF
                        T > Tmax
                        Bill ((|(T-Tmax)|/10)+1) * p * h
```

Aforementioned example could also be used in the context of a late delivery of the goods. Moreover, in the insurance sector, several parametric insurance applications might be constructed with a similar mechanism. Parametric insurance was already explained in Title IV.[353] In the event a hurricane arises, the parametric insurance policy holders will be entitled to claim their insurance payments, disregarding the extent of their suffered damages. The amount of such insurance payment will vary according to the magnitude of the hurricane. By analogy, in flight delay insurances, the overall flight delay time can determine the amount to be paid to the insurance policy holder.

In Title IV, we already discussed the Internet of Things and their potential application for smart contract use cases.[354] The growing interconnectivity of devices might affect the accuracy of oracles as well. Several integrated sensors can confirm the same events. For example, both the cargo ship sensors and the port sensors can register the unload of a container on the quays. By combining these oracles, the margin for error decreases, safeguarding the integrity of the transaction.

153. The described examples require a rather straightforward oracle input. The issues which are dealt with are easily quantifiable. With regard to the delivery of a cargo, the oracle only has to register

[353] *Supra,* 89: For more, see A. COHN e.a., "Smart after all: Blockchain, smart contracts, parametric insurance, and smart energy grids", *supra* from 213, 2017, vol. 2, 293.
[354] *Supra,* 85.

whether or not the goods have arrived. This resembles to the Boolean logic as described before (i.e. a value being either true or false).[355] The same goes for the example of the temperature sensors: either the temperature is too high or too low, or it is between those predetermined boundaries.
Things get more complicated when a more nuanced oracle input is required. An oracle could, for example, provide data on the quality of certain goods, or the extent of damages inflicted to the goods. Also, in the situation in which a consumer invokes the right to withdraw from a distance contract, the seller could easily install an oracle which automatically detects the returned goods and reimburses the consumer. However, the oracle should also be able to identify any diminished value caused by the consumer. In such cases, computational strategies need to be applied to divide the complexity of the issue into multiple sub-issues. By resolving these issues and combining the answers, accurate information can be fed into the smart contract, leading to the desired execution.
We already emphasized the right of the buyer to check the delivered goods for their conformity. Automating such process requires two steps to be taken. In step one, the buyer has to predetermine the conditions which should be met by the goods with respect to their conformity. To some extent, such exercise might align the perceptions of the buyer and the seller on the characteristics of the goods. In the Frigaliment case, as described above, buyer and seller might have agreed on the same type of chickens if their agreement was concluded in a smart contract form, because they had to specify precisely when the chickens would be considered conform to the agreement .[356] Subsequently, in step two, parties should implement oracles that can identify whether the goods meet these conditions. Evidently, different types of goods require different aspects to take into consideration. For example, checking the conformity of a cargo of cars will need to rely on different data as compared to checking a cargo of bananas. Checking the former will be a more complex technical operation, because the requirements to declare the cars' conformity will be more stringent. It will be, for example, rather difficult to judge the integrity of the cars' bodyworks. The nature of certain goods therefore decreases the opportunities for parties to establish an accurate virtual representation.

154. We can conclude that, in the situation where a smart contract has to rely on a more nuanced data input, generated by one or more oracles, the accuracy of such input is correlated to the nature of the goods. Furthermore, based on foregoing rationale, we can suggest several criteria that may indicate a higher or lower degree of complexity when comparing the condition of the physical object with its virtual representation:

- *High value.* Parties might be raising more concerns with regard to the conformity of high value goods, such as large vehicle cargo's, medicines or chemicals.

---

[355] *Supra,* 103.
[356] *Supra,* 120.

- *The amount of measurable characteristics of the good(s).* For example, a certain chemical raw material can be easily measured in terms of its chemical composition and volume, when assessing its quality and conformity. On the other hand, it is harder to determine whether the shipment of tailored, customer-specific clothes is conform to the agreement.
- *A great variety between individual goods in a cargo.* Uniformity among the goods enhances the ability to create a virtual representation of the goods.
- *The resources of both parties.* Evidently, large distribution companies will have more resources to establish an integrated oracle network. Consumers will most likely not be able to provide machine-based oracles themselves.

ii. A sufficient level of reliability

155. We discussed the necessity of machine-based oracles that can provide sufficient accuracy to create a virtual representation of physical objects. However, a second important consideration should be made when determining the effectiveness of machine-based oracles. As mentioned before, oracles are a powerful tool to broaden the scope of application for smart contracts. However, even though blockchain ledgers are considered immutable and tamper-proof, and by consequence, the smart contracts built on top of these blockchains as well, oracles can completely undermine this key asset. They can re-enter distrust in the system for the following reasons.

Machine-based oracles do not rely on any human intervention, and could therefore be considered tamper-proof. Nonetheless, one should be sceptical towards such assumption. If one of the parties is financially incentivised to manipulate the machine-based oracle, the reliability of the system can be at risk. Before, we gave the example of the transport of temperature sensitive goods. The oracle generating such temperature data would be a sensor in the truck. It is likely that the supplier has installed the temperature sensor in his truck (i.e. the oracle is under his control). A malicious supplier could put the sensor in a cooled compartment of the truck, thereby bypassing the obligation of keeping the entire truck refrigerated. In an attempt to save costs, suppliers might be tempted to attack the system. In other words, even though the temperature sensor is technically capable of providing accurate information on the temperature of the goods, it is no longer reliable.[357] Blockchain is designed to create a relationship of trust between the participants of the network. Paradoxically, the buyer has to trust the seller that he has not tampered with the oracles under the seller's control. This eliminates the initial foundation of trust created by the blockchain ledger and the smart contract.

On the other hand, if the buyer controls the machine-based oracle which registers the delivery of the goods, he can tamper with the oracle to avoid the automated execution of the smart contract (i.e. avoid the payment for the delivered goods). The buyer can, for example, shut down the oracle on the moment of the delivery.

---

[357] K. WÜST e.a., "Do you need a Blockchain?", *supra* from 55, 2018, 50.

Such trust issues are less problematic for oracles that are controlled by the party which benefits from honest behaviour. As we explained before, an escrow mechanism might provide the right incentives. Apart from these additional incentivising mechanisms, there is no clear resolution to aforementioned paradox.[358]

156. Up until now, we have been discussing the functionality of machine-based oracles. However, humans can provide data input in the smart contract as an oracle as well. Reliability issues are even more prominent for human-based oracles. Therefore, particular attention should be devoted to the internal incentives of each party to act in an honest manner. The escrow mechanism that was explained before already indicated how each party should be encouraged not to tamper with the integrity of the blockchain. The buyer will confirm that the goods are delivered, acting as a human-based oracle, because he will receive the stake he already inserted into the escrow mechanism in the smart contract. This illustrates the importance of incentivising parties when they are empowered to feed data into the smart contract as an oracle.

157. Apart from parties being oracles themselves, they can also agree on allocating such power to a trusted third party (hereafter: TTP). Such TTP can be used to create a multi-signature transaction. The three parties in such transaction are the buyer, the seller and the TTP. When two out of three sign the transaction (i.e. confirm to the smart contract that the good has been delivered), it is automatically executed. The buyer and the TTP will be entitled to sign the transaction once the seller has delivered the good. If the seller already signs the transaction before he delivers the good, he will need another signature to execute the transaction. If the buyer attempts to be dishonest and refuses to sign the transaction once the good has been delivered, the TTP can sign the transaction and execute the smart contract. Evidently, such mechanism heavily relies on the trustworthiness of the TTP.

c. *Integration in the new contract model*

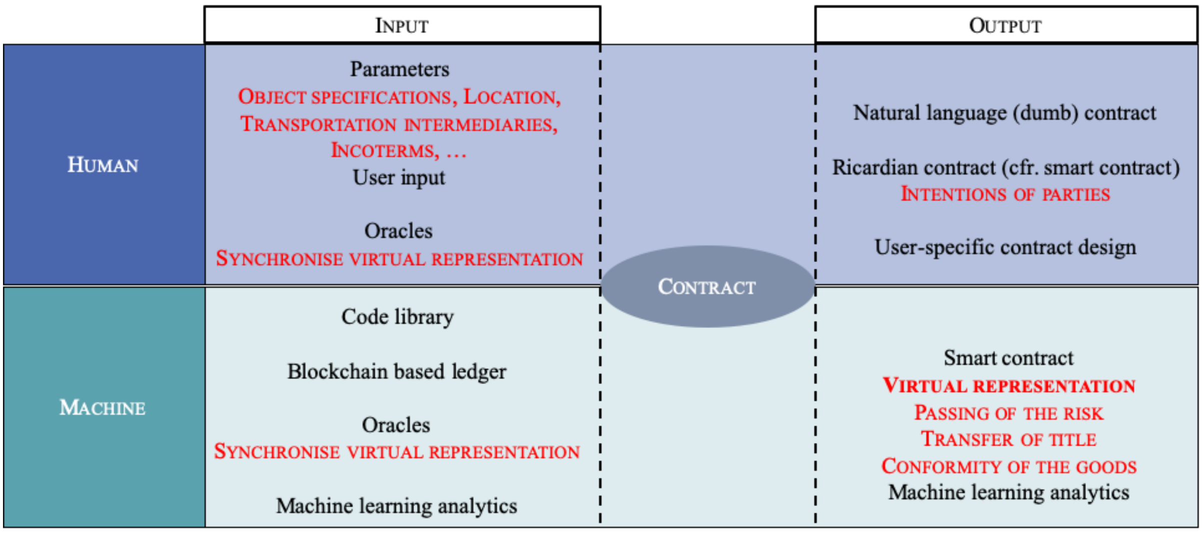
158. To recapitulate, the specifics of a physical object should be adopted in the smart contract as a virtual copy. Such a virtual copy describes the object's location, its state, its condition, its current owner, etc. Both the physical and the virtual copy should stay identical to safeguard the integrity of the agreement. Based on such synchronisation, parties can adopt smart contract clauses related to the physical object. They can determine that the risk of damages and/or destruction of the good automatically passes at a certain moment in time, they can automatically declare the goods conform to

---

[358] In some cases, an additional oracle might prevent the seller from tampering with the oracle that determines the temperature. For example, the temperature oracle can incorporate an additional oracle during the transport of a chemical liquid which needs to be preserved at a certain temperature. The additional oracle, while being linked to the temperature sensor, checks the amount of liquid in the supply tank of the truck. Hence, the combination of the oracles guarantees that a certain amount of chemical liquid is at a certain temperature.

their concluded agreement, they can determine the event that triggers the payment, they can adjust the price according to predetermined changes in the good's condition, etc. Nonetheless, if the synchronisation between the physical object and the virtual copy is either impossible or possible however interrupted by malicious behaviour, the agreement loses its integrity.

159. Firstly, we can conclude that B2B contracts dealing with the sale of those goods which show a great consistency in specifications, and which are sold on a large scale, can be the subject of smart contracts. This analysis is based on the possibility of integrating machine-based oracles that can enable an accurate and reliable synchronisation between physical objects and their virtual representations. We assume that parties which are already in a relationship of mutual trust can use smart contracts to optimize their transactions. On the contrary, parties that are less likely to trust each another, yet desire to adopt a smart contract, need to find recourse to other mechanisms. Such parties need to be incentivised not to tamper with the machine-based oracles under their own control. This can be achieved by either adding additional oracles that are controlled by a trusted third party or the counterparty, either by adding an escrow mechanism to the transaction. Also, a multi-signature relationship might resolve these trust issues by allowing a trusted third party to sign the transaction.
Assuming an accurate and reliable virtual representation of the goods is possible, legal consequences can be attached to such representation in the smart contract. In such scenario, the clauses dealing with the passing of the risk, the transfer of the title and the conformity of the goods can be adopted in the smart contract.



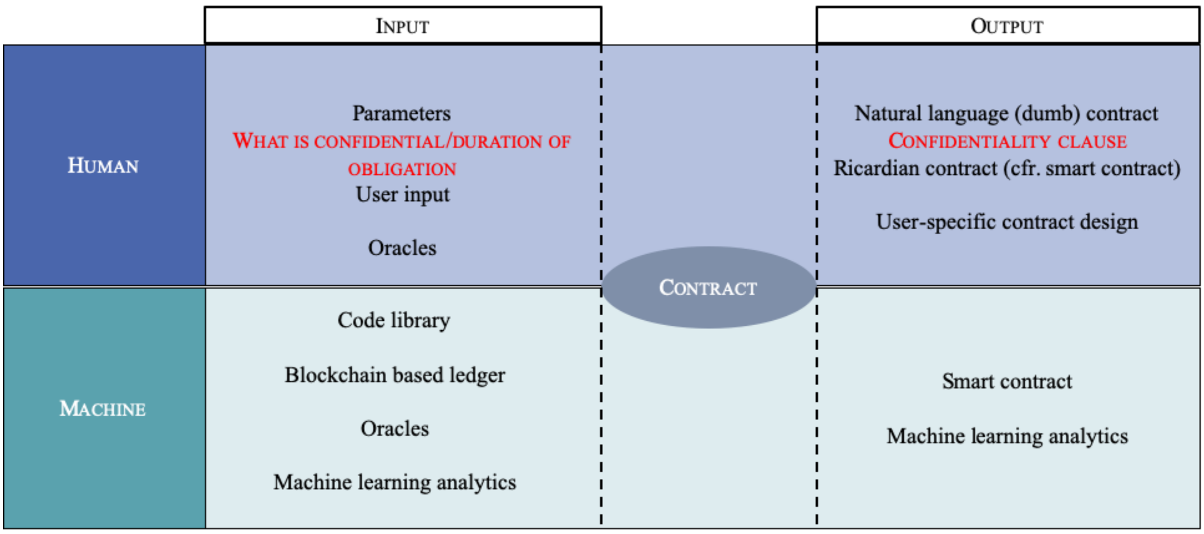
3.4. Boilerplate clauses

160. Boilerplate clauses refer to clauses which do not specify essential aspects of the contract, such as the price and the merchandise, but are adopted in the vast majority of commercial contracts to enhance the relationship between the contracting parties. Often, these clauses are standardised, as they are routinely included in contracts. This paragraph will give several prominent illustrations of boilerplate clauses, such as a force majeur clause. Again, these clauses will be mapped on the new contract model as described in Title IV.

a. Confidentiality clause

161. A confidentiality clause is adopted in many types of contracts. It creates an obligation for parties not to disclose any information expressed in the agreement. Either the entire agreement is considered confidential, either parties only declare certain clauses non-disclosable. The clause can also contain a certain period of time during which parties are bound to be confidential.

162. Confidentiality clauses should be distinguished from the confidentiality of the contract as such. Public blockchains might not be an adequate platform for smart contracts if the parties wish to keep their agreement confidential, as all users are entitled to right to read the transactions on the blockchain. Private blockchains might be a more adequate solution to the issue of confidential contracts. Nonetheless, confidentiality clauses in commercial contracts target the contracting parties, as opposed to third parties. Limiting the ability of disclosing information by one of the contracting parties goes beyond the capacities of any human- or machine-based oracle. Moreover, it is merely impossible for a oracles to identify a breach of the non-disclosure obligations. We can conclude that confidentiality clauses are not qualified for smart contract implementation. They should be adopted in the natural language (dumb) contract instead.



b. *Assignment clause*

163. Assignment clauses determine which contractual rights (i.e. claims) can be transferred to third parties on behalf of one of the contracting parties. For example, a buyer and a seller agree on the sale of a certain good. The buyer owes the seller €100. The seller is also a debtor of a third party. The seller might consider assigning the claim of €100 on the buyer to the third party to (partially) repay his own debt. The general rule is that all rights to performance are assignable.[359] However, both the law and the contract itself can prohibit such assignment. It is allowed to assign a monetary obligation in part.[360] Non-monetary obligations can only be partially assigned if the debtor consents to the agreement or if the assignment does not render the obligation significantly more burdensome.[361]
A relevant principle of European contract law is the effect of a contractual prohibition on assigning contractual rights: "A contractual prohibition of, or restriction on, the assignment of a right does not affect the assignability of the right."[362] However, if one of the parties breaches the contractual prohibition, the debtor can still perform in favour of the assignor and is discharged by so doing, unless the debtor has given his consent to the assignment.

164. According to the principles of European contract law, a contractual prohibition on assigning a right does not affect the ability to assign a right by either party. When taking this into consideration, several situations can be distinguished:

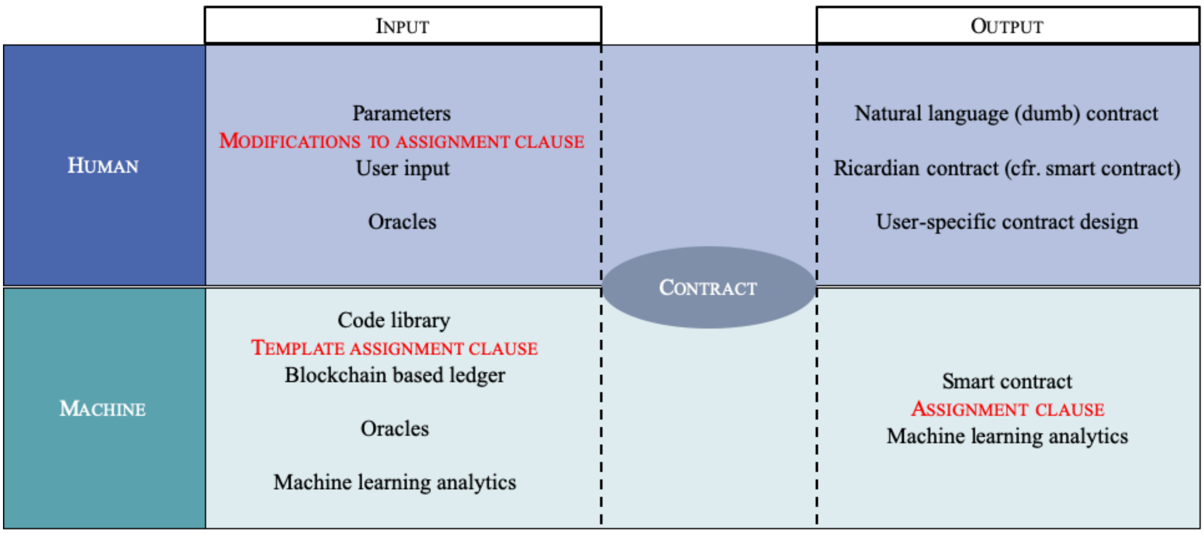
- *No assignment clause is adopted in the smart contract.* In a natural language contract, both parties will be entitled to assign their rights, based on the general principle of assignability of rights. However, as mentioned before, smart contracts are essentially immutable. In other words, after the conclusion of the agreement in smart contract form, no further adaptions can be made. Upon execution of the smart contract, the payment will be addressed to the initial counterparty. This leads to the conclusion that, without sacrificing the immutable character of smart contracts, no assignment of a right is possible if the smart contract does not provide in such possibility.
- *An assignment clause is adopted in the smart contract.* In case parties agree on including a provision in the smart contract that allows the assignment of rights, such provision can be linked to an oracle input. A party can provide such oracle an input when it wants to assign the right, and consequently, the specifications of the wallet of a third party can be inserted

[359] C. VON BAR e.a., *Principles, definitions and model rules of European private law: draft common frame of reference (DCFR)*, *supra* from 18, 2009, 469.
[360] *Ibid.,* 472.
[361] *Ibid.*
[362] *Ibid.,* 474

in the code to exercise monetary rights. For non-monetary rights, for example the delivery of certain goods, the oracle input can insert an additional location and identity specifications of the assignee. The smart contract can execute accordingly.

- *An explicit prohibition is adopted in the smart contract.* An explicit prohibition is, according to the DCFR, insufficient to prevent one of the parties from assigning their rights. Therefore, such provision would have the same legal effect in a smart contract as a smart contract without such provision at all. In other words, the smart contract would not allow parties to assign their rights, due to its immutable nature.

165. We can conclude that assignment clauses might be problematic to be implemented in smart contracts. Both the omission and the prohibition of an assignment clause lead to the same result. A couple of solutions can be proposed. Firstly, the immutability of smart contracts could be lifted. However, this would require the consent of all the nodes in the blockchain network. For smart contracts developed on Ethereum, this is not a valid option. Even for private permissioned blockchains, the technical complexity of such operation would not be worth the efforts. Secondly, the code library of the blockchain ledger could provide a smart contract template of a clause which enables assignment. When parties agree on drafting a smart contract, the possibility of assigning rights is automatically included in the contract. As a consequence, there will be an assignment clause in every smart contract, which parties then can modify to their preferences.



c. *Impediment clause*

166. As a fundamental principle, parties are bound to perform the obligations which arise from a concluded agreement. The non-performance of such obligations leads to the liability of the non-performing party. However, it might become impossible for a debtor to perform his obligation during the course of the agreement. Such impossibility might refer to a factual impossibility (e.g. the goods were burnt down), a practical impossibility (e.g. the delivery of a ring that has sunk to the bottom of the ocean) or a moral impossibility (e.g. the actor that should perform while his wife is dying at home).[363] the DCFR states that: "A debtor's non-performance of an obligation is excused if it is due to an impediment beyond the debtor's control and if the debtor could not reasonably be expected to have avoided or overcome the impediment or its consequences".[364] TJONG TJIN TAI distinguishes two steps that need to be taken in order to assess the existence of impediment: In step one, the cause of the non-performance needs to be found. In step two, one has to determine whether such cause is attributable to the debtor.[365]

i. First step: The cause of non-performance

167. There are a significant amount of possible causes that prevent the debtor from performing his contractual obligation. Such causes can be specific to each situation. For example, a debtor may have been declared bankrupt. However, it is also possible that the debtor was interrupted by an internet breakdown during the execution of his obligation (i.e. an online payment). Determining the cause of non-performance is therefore a complicated undertaking from a code perspective. Including an exhaustive list with possible causes in the smart contract is simply impossible. Two solutions are proposed to resolve this issue.[366]

168. The first solution is the adoption of a general impediment statement in the smart contract code, which is linked to a human-based expert oracle. Such oracle takes the form of an alternative dispute mechanism. Once the smart contract registers that a party does not perform its obligation (e.g. the seller does not deliver the goods within the agreed period of time), the non-performing party can invoke the expert oracle. Subsequently, the expert will assess the cause of non-performance. If he can identify a legitimate cause for non-performance, he can feed such data in the smart contract, redeeming the debtor from its duty to perform.

169. The second solution starts from the presumption that one of the parties should bear the risk of non-performance.[367] The risk bearing party is only excused when it can invoke one of the foreseeable causes that are explicitly adopted in the smart contract. Especially in a B2B context, parties are familiar with business risks and have the means to mitigate these risks, such as dedicated insurance policies. These causes can be linked to human-based or machine-based oracles. For example, the registration of a flood might excuse the debtor from delivering the destructed goods, if such cause is adopted in the

---

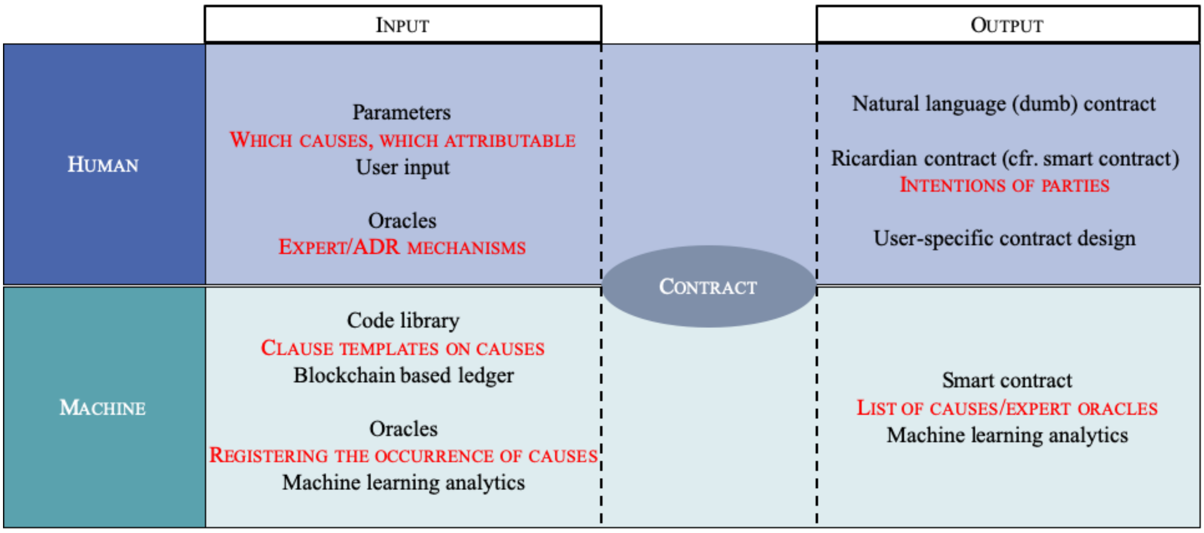
[363] E. TJONG TJIN TAI, "Force Majeure and Excuses in Smart Contracts", *supra* from 238, 2018, 7.
[364] C. VON BAR e.a., *Principles, definitions and model rules of European private law: draft common frame of reference (DCFR)*, *supra* from 18, 2009, 357.
[365] E. TJONG TJIN TAI, "Force Majeure and Excuses in Smart Contracts", *supra* from 238, 2018, 12.
[366] *Ibid.,* 12-13.
[367] *Ibid.*

smart contract and a machine-based oracle can identify the intensity of the flood. Nonetheless, not every cause adopted in the smart contract is directly linked to the non-performance of a debtor. There could be, for example, an underlying earlier cause. We can conclude that, to effectively include a mechanism dealing with impediment in the smart contract, a thorough research should be done on possible causes for non-performance. The code library of a blockchain network might provide clause templates on different causes, which are already linked to machine-based oracles (e.g. the registration of electricity shutdowns).

ii. Second step: The attributability of the cause

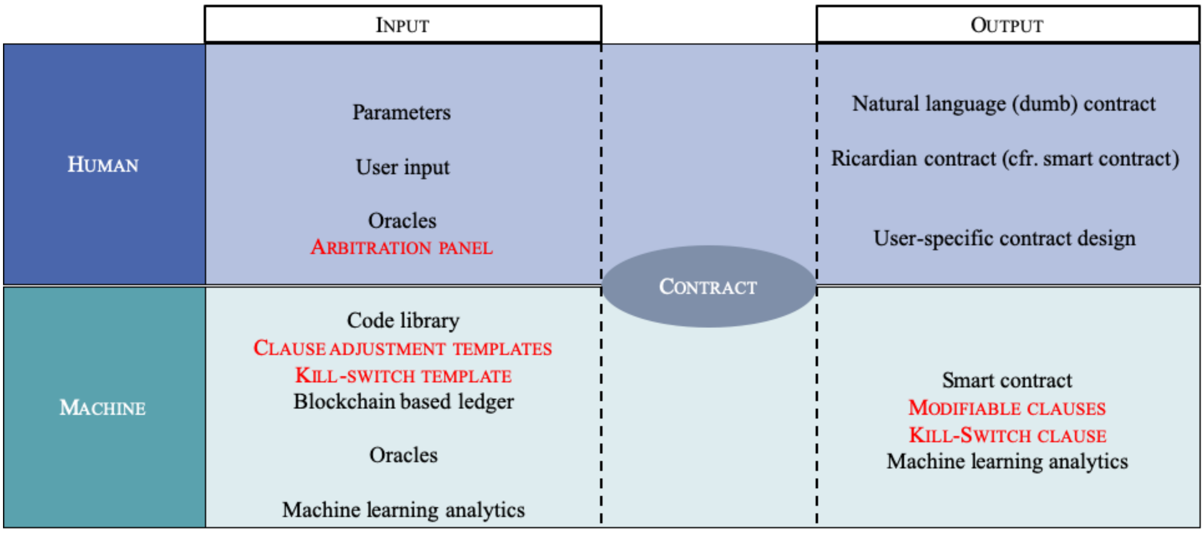
170. In the aforementioned example of the flood that destroyed the goods of the debtor, one could question whether or not the flood is attributable. The debtor could, for example, have kept the goods in a leaking basement without any protection against liquid damages. In this case, the goods were probably destructed by the debtors negligence and poor caretaking of his goods. Something is attributable when it explains or indicates a cause. There should be a direct connection between the cause and the result (e.g. the mere existence of a flood will not always be the cause of non-performance).
Attribution of cause is a delicate exercise, as it is an open norm and requires further interpretation. This can be remedied by declaring certain causes valid as an excuse for non-performance in any case. In modern day commercial contracts, parties already agree on causes which do or do not form a legal basis for non-performance of an obligation. The same practices would be implemented in the smart contract. To determine the attribution of cause, parties could also opt to seek recourse to an expert as a human-based oracle as well.

171. When combining the aforementioned two steps, a conclusion can be drawn for the implementation of impediment clauses in a smart contract. On the one hand, parties can opt for a general rule on impediment, which is maintained by using human-based oracles as experts. On the other hand, parties can also implement a list with possible causes. The occurrence of such clauses is identified by human- and machine based oracles. Attribution of those causes is presumed. Evidently, parties can opt for a combination of these solutions.



d. Arbitration clause

172. We discussed the relevance of alternative dispute mechanisms on several occasions during this research. Arbitration clauses embody the agreement of parties to settle future disputes outside of court, through an arbitration process. Although they are not yet in existence, specialised arbitration panels might be constructed, which solely focus on resolving smart contract disputes. Ideally, the members of the panel are familiar with program languages and the limitations of code in contracts. Also, they need to be aware of their function as 'oracle'. Their decision will be fed into the smart contract and, consequently, affect the remaining execution of the contract. In this regard, it should be noted that parties might benefit from inserting a so-called "kill-switch" in the smart contract, which can be triggered by the arbitration panel, if necessary. The kill-switch terminates the smart contract immediately. Such termination should not be considered an obvious possibility. As mentioned before, smart contracts are immutable and will not stop executing without any predetermined line of code which interrupts execution. The kill-switch function, and the relating oracle input (i.e. the decision of the arbitration panel) can be adopted in the code library as a template. Similar to a kill-switch, parties may indicate which smart contract clauses should be modifiable by an arbitration panel.



e. Good faith (clause)

173. Some principles do not need to be externalised in contracts to have legal implications. They affect every contractual relationship. Such 'implicit clauses' impose duties on contracting parties to behave or not to behave in a specific manner. Some implicit clauses also entitle a party a certain right. We will briefly discuss the principle of good faith.

174. Good faith is a duty which is imposed on every party that enters into an agreement. The DCFR stipulates that: "person has a duty to act in accordance with good faith and fair dealing in performing an obligation, in exercising a right to performance, in pursuing or defending a remedy for non-performance, or in exercising a right to terminate an obligation or contractual relationship.".[368] It is further specified that the principle of good faith cannot be excluded or limited by contract.

175. Good faith refers to the acts and omissions of parties during the course of the agreement.[369] Similar to the presented solutions for the issue of smart impediment clauses, one could argue that the smart contract might adopt a list of acts or omissions which are ought to infringe the principle of good faith. However, whereas excuses for the non-performance of obligations can be specified to a certain extent, acts and omissions violating the principle of good faith are not. Theoretically, a party could violate the principle by invoking any right of the contract in disproportionate manner, as opposed to the non-performance of obligations only. Therefore, the scope of application of the principle of good faith is significantly larger than the scope of application of the theory of impediment. Even tough parties

---

[368] C. VON BAR e.a., *Principles, definitions and model rules of European private law: draft common frame of reference (DCFR)*, *supra* from 18, 2009, 303.

[369] The principle of good faith will also have legal implications during the phase of negotiation. We will only focus on the implications after the conclusion of the contract.

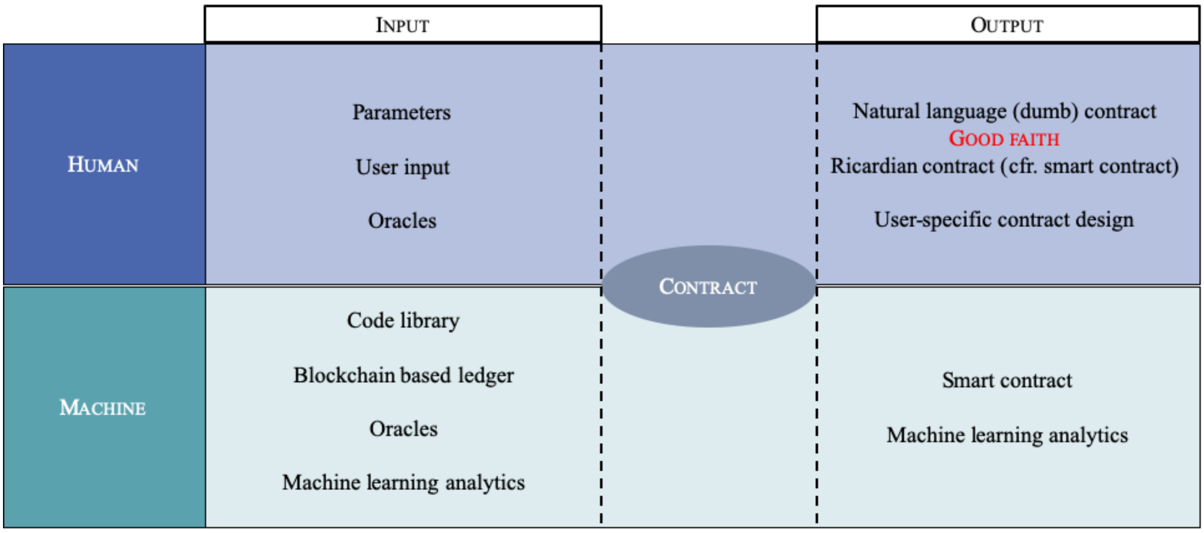
would attempt to list acts and omissions which would violate the principle of good faith, one cannot assume that one of the parties will bear the risk for the non-listed violations.
On the contrary, one could argue that the opportunities for parties to engage in behaviour violating the principle of good faith are limited, as the smart contract executes itself. Several actions do no longer need active participation of parties, but are automated. Nonetheless, good faith cannot be completely ignored.

176. Two other solutions are possible. Firstly, an human-based oracle could be inserted in the smart contract, functioning as an expert which determines the behaviour of parties. Such expert would threaten the autonomy of parties during the execution of the contract, as he will have the ability to intervene in the smart contract code. As mentioned before, the principle of good faith affects the contract as a whole, implying legal consequences for every contract clause. Unlike an expert intervention in the smart contract clauses that deal with impediment, every clause will be subject to scrutiny by the expert analysing possible violations of the principle of good faith. Parties might use such opportunity to contest certain actions for tactical reasons or reasons of self-interest. In addition, the impact of the principle of good faith on commercial transactions is rather limited.[370] Parties are usually aware of professional commercial conduct and want to maintain their credibility and relationships with partners. `
In other words, by implementing a human-based oracle which can affect every right and obligation in the smart contract, the immutable character of the smart contract might be compromised.

177. With the foregoing in mind, we propose a second solution that does not compromise the self-executing abilities of smart contracts. Such solution would propose to rely on courts to assess the compliance with the principle of good faith. We argue that smart contract provisions are, after being drafted and deployed on the blockchain, beyond the control of parties. Such clauses can only be executed in a single manner: through self-execution. The CPU itself cannot alter how smart contract clauses are executed. Therefore, parties are, or should be, aware on the legal consequences of the smart contract clauses they adopt. Any conduct violating the principle of good faith is likely to reside outside of the smart contract. We can conclude that the conventional courts and tribunals are the most suitable and efficient dispute resolution mechanisms to scrutinize such conduct. Such solution maintains the self-executing character of smart contracts, while acknowledging the importance of the principle of good faith. The natural language contract can refer to the competence of courts and tribunals in the assessment of the principle of good faith.

---

[370] C. VON BAR e.a., *Principles, definitions and model rules of European private law: draft common frame of reference (DCFR)*, *supra* from 18, 2009, 304.

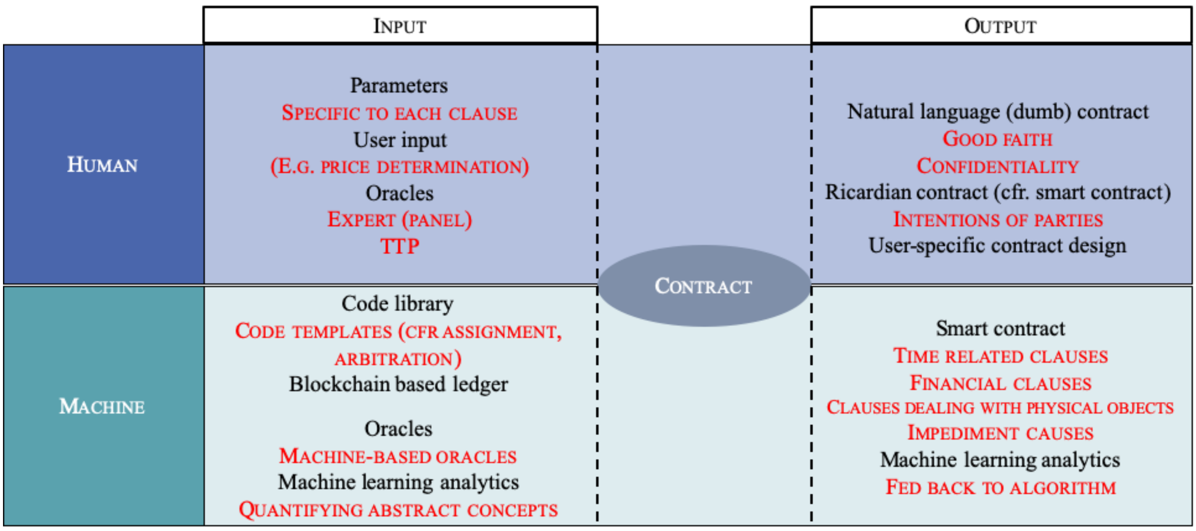

INPUT
OUTPUT
HUMAN
Parameters
User input
Oracles
Natural language (dumb) contract
GOOD FAITH
Ricardian contract (cfr. smart contract)
User-specific contract design
CONTRACT
MACHINE
Code library
Blockchain based ledger
Oracles
Machine learning analytics
Smart contract
Machine learning analytics

CHAPTER 4. CONCLUSION

178. This Title was the final step in a three-fold analysis. Chapter one gave a brief insight in the characteristics of computer programs and computer language. We explained the binary nature of machine code and its implications towards ambiguity.

Subsequently, chapter two emphasised the existence of vague concepts in contracts. We concluded that parties, when drafting smart contracts, should be aware of the necessary adaptations in the expression of their intentions. The CPU cannot cope with natural language expressions, or their coded translation, such as "promptly". To a certain extent, smart contracts might enhance the harmonisation of the intentions of parties, as they need to eliminate such ambiguous terms. Chapter two also provided a general overview on several mechanisms to quantify abstract concepts, such a reasonableness. Modern technologies, such as machine learning, might play a significant role in achieving such translation. Also, the practical implications of oracles were discussed. We raised some concerns with regard to human-based oracles and their impact on the automated execution.

Finally, chapter three attempted to apply the knowledge that has been accumulated throughout this research to frequently drafted contract clauses, with a particular focus on commercial contracts. We distinguished four categories of clauses: time-related clauses, financial clauses, clauses relating to physical objects and boilerplate clauses. Such clauses were analysed in the light of the principles of European contract law with regard to their legal significance and implications. As a result, these clauses were mapped in the new contract model. To summarize our findings, we can provide the following schematic:



The schematic indicates the importance of a hybrid contract model. Both natural language contract and smart contract (with the attached Ricardian contract) will need to be perfectly coordinated. Inevitably, the implementation of smart contracts will add technical and legal complexity to contracts as we know them. This research does not aim to answer questions with regard to the cost-effectiveness of such added complexity. However, we can conclude that, to a great extent, contract clauses can be translated to smart contract clauses. Such translation assumes the availability of several building blocks, which together form a new contract model.

a. Input

179. The human input is mainly delivered through the adjustment of contract specific parameters. Examples are the price, specifications of the merchandise, etc. These parameters are connected to the code library, in which templates of coded clauses are stored. Such approach has two consequences: the threshold for parties to adopt smart contracts is lowered, as they do not need to be able to write code themselves. In addition, unconventional intentions by parties cannot be expressed in code by relying on the code library templates. In such situation, either parties adopt a standard commercial contract in natural language, either they insert coded clauses themselves.

During the execution of the agreement, both the contracting parties (i.e. user input) as human-based oracles can provide an input in the contract for multiple purposes. One example is the input from an expert panel, settling a dispute concerning a case of impediment.

180. On the machine side, we already explained the code library. A blockchain based ledger might preserve the integrity of concluded smart contracts. Blockchain technology ensures the immutability of smart contracts and reduces the risk on malicious attacks. However, as mentioned before, we argue that a blockchain ledger is not a requirement as such to draft valid smart contracts. Similar to human-based oracles, smart contracts can provide in data entry possibilities by machine-based oracles. An example is the sensor in the loading bay of a port, detecting incoming and outgoing shipments. Finally, machine learning algorithms can be enabled to support parties in their decision making, and to quantify abstract concepts such as "a reasonable period". Again, the effectiveness of such algorithms heavily relies on the amount of comparable cases as a point of reference.

b. Output

181. Three output channels serve as a human interpretation of the contract. The first channel is the natural language contract, which contains those clauses that are not adopted in the smart contract. Their enforcement happens through the conventional institutions (i.e. the courts and tribunals). We argued that confidentiality clauses and the enforcement of the principle of good faith belong in the natural language (dumb) contract. Either the described mechanisms to quantify abstract concepts are inappropriate to codify these clauses, either the implementation in smart contracts would compromise the immutability of smart contracts in a disproportionate manner. The second channel refers to the Ricardian contract. As explained in Title IV, the Ricardian contracts adopts the intentions of parties which are expressed in the smart contract. The Ricardian contract might form a means of interpretation in the case of a rising dispute between parties. It reflects the parameters which are adjusted by parties in the input section. Finally, the contract also provides a third human output channel in the form of user-specific contract design. This research did not emphasize the potential applications for such output channel.

182. Apart from human-based output, the contract model also provides in a machine based output. The coded clauses of smart contracts are directed towards the CPU which can execute them. Our research showed that several important types of clauses can be put in the smart contract. Time related clauses, financial clauses (i.e. monetary obligations), clauses dealing with physical objects (e.g. the passing of the risk of destruction) and several boilerplate clauses qualify for a smart contract implementation.

TITLE VI. FINAL CONSIDERATIONS

183. This research attempted to clarify whether or not contract clauses can be transformed to smart contract clauses. The answer to this question is most definitely affirmative. Today's technologies enable the transformation of a significant amount of contract clauses, which eventually will be able to execute themselves. Even though smart contracts are still a buzzword. Also, they are overestimated by many. However, they do have a substantial potential to enhance trust and efficiency in the contractual relationship of parties. It should be recognized that there are still many remaining steps to be taken. To successfully integrate a new contract model, one should have examined accurate projections on its technical and legal impact.

184. During the course of this research, we discovered several potential pitfalls that might stall commercial integration of smart contracts. Some of these issues require further research. Among others, we consider following topics to be examined for the purpose of commercial integration of smart contracts:

- Smart contracts will heavily rely on oracles. As explained before, oracles can reintroduce distrust into the relationship between parties. Parties may not be able to tamper with blockchain based smart contracts, but they could be incentivised to manipulate the oracles which are connected to the smart contract. Such a paradox of trust should be resolved.
- We pointed out that a code library, providing several templates of coded clauses, might be a solution to lower the threshold for parties to adopt smart contract systems. Such library needs to be drafted extremely carefully by a cooperation of lawyers and programmers. Carelessness and inconsistency in these templates might lead to so-called 'smart bugs', which can have a huge impact on parties' agreements (e.g. contracts that cannot be terminated and keep on executing for ever).
- This research mainly focused on the integration of commercial contract clauses. Other industries might be characterised by more specific clauses which could be more problematic to translate to coded clauses.
- Legislations are still unclear on the legality of smart contracts. A recommendation on a legislative framework might be the right incentive towards the commercial viability of smart contracts.

TITLE VII. BIBLIOGRAPHY

LEGISLATION AND LEGISLATIVE DOCUMENTS

European Union

Directive 2001/83/EC of the European Parliament and of the Council of 6 November 2001 on the Community code relating to medicinal products for human use.

Directive 2011/83/EU of the European Parliament and of the Council of 25 October 2011 on consumer rights.

United States

Electronic Transactions of the Arizona Revised Statutes.

Proposal of legislation HB5553 of the Illinois General Assembly, 8th of January 2019.

Belgium

Belgian Civil Code.

GUIDANCE PAPERS AND STATEMENTS

Guidelines (Comm.) of 5 November 2013 on Good Distribution Practice of medicinal products for human use, *Official Journal of the European Union,* 23 November 2011, 2013/C 343/01.

CASE LAW

Belgium

Cass. 22 November 1973, *Pas.* 1974, I, 312.

United States

Frigaliment Importing Co. ,Ltd. ,v. BNS International Sales Corp., 190F.Supp.116, 117 (S.D.N.Y. 1960).

LITERATURE

Books

BASHIR, I., *Mastering Blockchain: Distributed ledger technology, decentralization, and smart contracts explained*, Birmingham, Packt Publishing Ltd, 2018.

BENTOV, I., GABIZON, A. and MIZRAHI, A., “Cryptocurrencies without proof of work” in *Financial Cryptography*, Springer, 2016.

BUES, M.-M. and MATTHAEI, E., “LegalTech on the Rise: Technology Changes Legal Work Behaviours, But Does Not Replace Its Profession” in *Liquid legal: Transforming Legal into a Business Savvy, Information Enabled and Performance Driven Industry*, Singapore, Springer, 2017.

BUYYA, R., DASTJERDI, A.V. and KHODADADI, F., “Internet of Things: An Overview” in *Internet of Things: Principles and paradigms*, Australia, Elsevier, 2016.

CORRALES, M., FENWICK, M. and HAAPIO, H., “Digital Technologies, Legal Design and the Future of the Legal Profession” in *Legal Tech, Smart Contracts and Blockchain*, Singapore, Springer, 2019.

DANNEN, C., *Introducing Ethereum and Solidity: Foundations of Cryptocurrency and Blockchain Programming for Beginners*, New York, Apress, 2017.

DE POTTER DE TEN BROECK, M., DE CORTE, R., DE GROOTE, B. and BRULOOT, D., “Verbintenis uit overeenkomst” in *Privaatrecht in hoofdlijnen*, Antwerpen, Intersentia, 2017.

DOWEK, G., *Principles of programming languages*, France, Springer Science & Business Media, 2009.

DRESCHER, D., *Blockchain Basics: A Non-Technical Introduction in 25 Steps*, New York, Springer, 2017.

FACK, V., *Algoritmen en datastructuren*, Leuven, Acco, 2011.

FRIEDMAN, D.P., WAND, M. and HAYNES, C.T., *Essentials of programming languages*, Cambridge, MIT press, 2001.

GERARD, D., *Attack of the 50 foot blockchain: Bitcoin, blockchain, Ethereum & smart contracts*, USA, CreateSpace Independent Publishing Platform, 2017.

GLASSBOROW, F., *You Can Do It!: A Beginner’s Introduction to Computer Programming*, West Sussex, John Wiley & Sons, 2004.

GRUNDMANN, S. and HACKER, P., “The Digital Dimension - The Architecture” in *European Contract Law in the Digital Age*, Cambridge, Intersentia, 2018.

HULL, J., TREEPONGKARUNA, S., COLWELL, D., HEANEY, R. and PITT, D., *Fundamentals of futures and options markets*, Australia, Pearson Higher Education AU, 2013.

KERNIGHAN, B.W. and RITCHIE, D.M., *The C programming language second edition*, New Jersey, Prentice Hall, 2006.

LANDO, O. and BEALE, H.G., *Principles of European contract law: Parts I and II*, The Hague, Kluwer Law International BV, 2000.

LESSIG, L., *Code version 2.0*, New York, Basic Books, 2006, http://codev2.cc/download+remix/Lessig-Codev2.pdf. (Consulted on 13 May 2019).

PAGALLO, U., CORRALES, M., FENWICK, M. and FORGÓ, N., "The Rise of Robotics & AI: Technological Advances & Normative Dilemmas" in CORRALES, M., FENWICK, M. and FORGÓ, N. (eds.), *Robotics, AI and the Future of Law*, Singapore, Springer, 2018, 1–13, https://doi.org/10.1007/978-981-13-2874-9_1. (Consulted on 13 May 2019)

PASSERA, S., "Beyond the wall of text: How information design can make contracts user-friendly" in *Design, user experience, and usability: users and interactions. Lecture Notes in Computer Science*, Cham, Springer, 2015.

ROCHKIND, M.J., *Advanced UNIX programming*, United States, Pearson Education, 2004.

SINGHAL, B., DHAMEJA, G. and PANDA, P.S., "How blockchain works" in *Beginning Blockchain: A Beginner's Guide to Building Blockchain Solutions*, Springer, 2018.

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

TREITEL, G.H., *The law of contract*, Sweet & Maxwell, 2003.

TUCKER, A.B., *Computer science handbook*, Florida, CRC press, 2004.

UNSWORTH, R., "Smart Contract This! An Assessment of the Contractual Landscape and the Herculean Challenges it Currently Presents for 'Self-executing' Contracts" in *Legal Tech, Smart Contracts and Blockchain*, Singapore, Springer, 2019.

VAN OEVELEN, A. and DE BOECK, A., "Overmacht en imprevisie in het Belgische contractenrecht" in *Overmacht*, Mortsel, Intersentia, 2015, https://www.jurisquare.be/en/book/9789400006911/overmacht-en-imprevisie-in-het-belgische-contractenrecht/ (Consulted on 27 March 2019).

VON BAR, C., CLIVE, E. and SCHULTE-NÖLKE, H., *Principles, definitions and model rules of European private law: draft common frame of reference (DCFR)*, Oxford, Oxford University Press, 2009.

WALLACH, W. and ALLEN, C., *Moral machines: Teaching robots right from wrong*, New York, Oxford University Press, 2008.

WILLOUGHBY, G., *Purebasic: A Beginner's Guide to Computer Programming*, Fegersheim, Aardvark Global Publishing, 2006.

Journals and online research papers

ALHARBY, M. and VAN MOORSEL, A., "Blockchain-based smart contracts: A systematic mapping study", *arXiv preprint arXiv:1710.06372*, 2017, https://arxiv.org/abs/1710.06372 (Consulted on 18 November 2018).

ANDROULAKI, E., BARGER, A., BORTNIKOV, V., CACHIN, C., CHRISTIDIS, K., DE CARO, A., ENYEART, D., FERRIS, C., LAVENTMAN, G. and MANEVICH, Y., "Hyperledger fabric: a distributed operating system for permissioned blockchains" , *ACM*, 2018, https://arxiv.org/pdf/1801.10228 (Consulted on 17 October 2018).

ATZORI, M., "Blockchain technology and decentralized governance: Is the state still necessary?", 2015, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2709713 (Consulted on 29 December 2018).

BERG, C., DAVIDSON, S. and POTTS, J., "The blockchain revolution", *The Institute of Public Affairs Review: A Quarterly Review of Politics and Public Affairs*, 2017, vol. 4, 34, https://search.informit.com.au/documentSummary;dn=342671732445019;res=IELHSS (Consulted on 3 January 2019).

BERGER-WALLISER, G., BARTON, T.D. and HAAPIO, H., "From visualization to legal design: A collaborative and creative process", *American Business Law Journal*, 2017, vol. 2, 347–392, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2841030 (Consulted 17 February 2019).

BHANDARI, B., *Supply Chain Management, Blockchains and Smart Contracts*, New York, NYU University of Law, 2018, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3204297 (Consulted on 18 March 2019).

BLAU, B.M., "Price dynamics and speculative trading in bitcoin", *Research in International Business and Finance*, 2017, vol. 41, 15–21, https://www.sciencedirect.com/science/article/pii/S0275531917303057 (Consulted on 12 November 2018).

BOCEK, T., RODRIGUES, B.B., STRASSER, T. and STILLER, B., "Blockchains everywhere-a use-case of blockchains in the pharma supply-chain", *2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*, 2017, 772–777, http://dl.ifip.org/db/conf/im/im2017exp/119.pdf (Consulted on 20 March 2019).

BOGNER, A., CHANSON, M. and MEEUW, A., *A decentralised sharing app running a smart contract on the ethereum blockchain*, ACM, Proceedings of the 6th International Conference on the Internet of Things, 2016, http://cocoa.ethz.ch/downloads/2017/08/2306_Sharing_App_Final_Publication.pdf (Consulted on 25 March 2019).

BRUNSCHWIG, C., “Multisensory law and legal informatics: a comparison of how these legal disciplines relate to visual law”, *Jusletter IT*, 2011, http://jusletter-it.weblaw.ch/en/issues/2011/104/article_324.html (Consulted on 4 March 2019).

CATALINI, C. and GANS, J.S., “Initial coin offerings and the value of crypto tokens”, National Bureau of Economic Research, 2018, https://www.nber.org/papers/w24418 (Consulted on 2 May 2019).

CHAWLA, H., *Blockchain: the key to anarchist self-governance?*, Warwick, University of Warwick, 2018, https://www.researchgate.net/publication/326190836_Blockchain_the_key_to_anarchist_self-governance (Consulted on 23 January 2019.

CHOHAN, U.W., *What is a Ricardian contract?*, Canberra, University of New South Wales, 11 December 2017, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3085682 (Consulted on 20 March 2019).

CHRISTIDIS, K. and DEVETSIKIOTIS, M., “Blockchains and smart contracts for the internet of things”, *IEEE Access*, 2016, 2292–2303, https://mycourses.aalto.fi/pluginfile.php/378344/mod_resource/content/1/Christidis%20and%20Devetsikiotis.pdf. (Consulted on 23 March 2019)

CLACK, C.D., “Smart Contract Templates: Legal semantics and code validation”, *Journal of Digital Banking*, 2018, vol. 4, 338–352, http://www0.cs.ucl.ac.uk/staff/C.Clack/research/JDigitalBanking-Clack-AuthorPreprint.pdf (Consulted on 26 February 2019).

CLACK, C.D., BAKSHI, V.A. and BRAINE, L., “Smart contract templates: foundations, design landscape and research directions”, *Position paper*, 2016, https://arxiv.org/pdf/1608.00771v2.pdf. (Consulted on 2 April 2019)

COHN, A., WEST, T. and PARKER, C., “Smart after all: Blockchain, smart contracts, parametric insurance, and smart energy grids”, *Georgetown Law Technology Review*, 2017, vol. 2, 273–303, https://georgetownlawtechreview.org/smart-after-all-blockchain-smart-contracts-parametric-insurance-and-smart-energy-grids/GLTR-04-2017/ (Consulted on 17 December 2018).

CONNOLLY, J. and COOKE, D., “The pragmatics of programming languages”, *Journal of the International Association for Semiotic Studies*, 2004, 149–162, https://www.degruyter.com/view/j/semi.2004.2004.issue-151/semi.2004.065/semi.2004.065.xml (Consulted on 8 April 2019).

CUCCURU, P., “Beyond bitcoin: an early overview on smart contracts”, *International Journal of Law and Information Technology*, 2017, vol. 3, 179–195, https://academic.oup.com/ijlit/article-abstract/25/3/179/3106217?redirectedFrom=fulltext (Consulted on 16 October 2019).

CUMMINS, T., "Commercial agility and creativity through contract simplification", Presentation at first international conference on contract simplification, Rüschlikon, 29 March 2016, http://media.swissre.com/documents/Presentation_Tim_Cummins.pdf (Consulted on 4 March 2019).

DE ANGELIS, S., ANIELLO, L., BALDONI, R., LOMBARDI, F., MARGHERI, A. and SASSONE, V., "PBFT vs proof-of-authority: applying the CAP theorem to permissioned blockchain", 2018, https://eprints.soton.ac.uk/415083/2/itasec18_main.pdf (Consulted on 7 March 2019).

DEMEYER, M., *Blockchain Technology And Smart Contracts From A Financial Law Perspective*, Ghent, University of Ghent, 2018, https://lib.ugent.be/fulltxt/RUG01/002/479/332/RUG01-002479332_2018_0001_AC.pdf (Consulted on 13 March 2019).

EUROPEAN UNION, "What if blockchain technology revolutionised voting?", *At a glance*, 2016, http://www.europarl.europa.eu/RegData/etudes/ATAG/2016/581918/EPRS_ATA%282016%2958191 8_EN.pdf (Consulted on 18 March 2019).

GANS, J.S., "The Fine Print in Smart Contracts", *SSRN*, 3 January 2019, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3309709 (Consulted on 4 March 2019).

GATTESCHI, V., LAMBERTI, F., DEMARTINI, C., PRANTEDA, C. and SANTAMARÍA, V., "Blockchain and smart contracts for insurance: Is the technology mature enough?", *Future Internet*, 2018, vol. 2, 20, https://www.mdpi.com/1999-5903/10/2/20/htm. (Consulted on 3 April 2019).

GEIREGAT, S., "Eigendom op bitcoins", *RW*, 2017, vol. 27, 1043–1050.

GIANCASPRO, M., "Is a 'smart contract' really a smart idea? Insights from a legal perspective", *Computer Law & Security Review*, 2017, vol. 6, 825–835, http://www.sciencedirect.com/science/article/pii/S026736491730167X. (Consulted on 21 October 2018)

GRIGGS, K.N., OSSIPOVA, O., KOHLIOS, C.P., BACCARINI, A.N., HOWSON, E.A. en HAYAJNEH, T., "Healthcare Blockchain System Using Smart Contracts for Secure Automated Remote Patient Monitoring", *Journal of medical systems*, 2018, vol. 7, 130, https://www.ncbi.nlm.nih.gov/pubmed/29876661 (Consulted on 4 March 2019).

GRIMMELMAN, J., "All Smart Contracts Are Ambiguous", *Penn Journal of Law and Innovation*, 2019, https://ssrn.com/abstract=3315703. (Consulted on 5 February 2019)

HAAPIO, H., PLEWE, D. and DEROOY, R., "Next Generation Deal Design: Comics and Visual Platforms for Contracting", *19th International Legal Informatics Symposium IRIS*, 2016, 373–380, https://www.researchgate.net/publication/299472189_Next_Generation_Deal_Design_Comics_and_ Visual_Platforms_for_Contracting (Consulted on 27 January 2019).

HSIEH, Y.-Y. and VERGNE, J.-P., "Bitcoin and the rise of decentralized autonomous organizations", *Journal of Organization Design*, 2018, https://www.ssrn.com/abstract=3082911 (Consulted on 12 March 2019).

IBM INSTITUTE FOR BUSINESS VALUE, "Device democracy: Saving the future of the Internet of Things", IBM, 2014, https://www.ibm.com/downloads/cas/Y5ONA8EV (Consulted on 14 March 2019).

JACCARD, G., "Smart Contracts and the Role of Law", *Justletter IT*, 2017, https://www.ssrn.com/abstract=3099885 (Consulted on 3 January 2019).

KATZ, D.M., "Quantitative legal prediction-or-how i learned to stop worrying and start preparing for the data-driven future of the legal services industry", *Emory LJ*, 2012, 909, https://heinonline.org/HOL/Page?handle=hein.journals/emlj62&div=27&g_sent=1&casa_token=S0xIbJI70RUAAAAA:0JUJMpSGeZ1P2kYOxuwtaQFu6wT591t6t5RYivgjG3SNTY7laEn6t_L3TyK_CP7Hnvunb7nt (Consulted on 18 April 2019).

KIM, E.E. and TOOLE, B.A., "Ada and the first computer", *Scientific American*, 1999, vol. 5, 76–81.

KOÇ, A.K., YAVUZ, E., ÇABUK, U.C. and DALKILIÇ, G., "Towards secure e-voting using Ethereum blockchain", *Proc. 6th International Symposium on Digital Forensic and Security (ISDFS 2018)*, 2018, https://www.researchgate.net/profile/Umut_Cabuk/publication/323318041_Towards_Secure_E-Voting_Using_Ethereum_Blockchain/links/5a931e2faca272140565c7e4/Towards-Secure-E-Voting-Using-Ethereum-Blockchain.pdf (Consulted on 17 March 2019).

KSHETRI, N., "Will blockchain emerge as a tool to break the poverty chain in the Global South?", *Third World Quarterly*, 2017, vol. 8, 1710–1732, https://www.tandfonline.com/doi/abs/10.1080/01436597.2017.1298438 (Consulted on 11 February 2019).

LUHMANN, N., "Das recht der gesellschaft", *Frankfurt am Main*, 1993.

MAATEN, E., "Towards remote e-voting: Estonian case", *Electronic Voting in Europe-Technology, Law, Politics and Society*, 2004, 83–100, https://pdfs.semanticscholar.org/ff4d/0a77e7561e62fd0258280c0baa02d8256a03.pdf (Consulted on 17 March 2019).

MACK, C.A., "Fifty years of Moore's law", *IEEE Transactions on semiconductor manufacturing*, 2011, vol. 2, 202–207.

MCCORRY, P., SHAHANDASHTI, S.F. and HAO, F., "A smart contract for boardroom voting with maximum voter privacy", *International Conference on Financial Cryptography and Data Security*, 2017, 357–375, https://www.researchgate.net/publication/317843497_A_Smart_Contract_for_Boardroom_Voting_with_Maximum_Voter_Privacy (Consulted on 18 March 2019).

MCGEVERAN, W., "Programmed Privacy Promises: P3P and Web Privacy Law", *NYU Law Review*, 2001, https://www.nyulawreview.org/issues/volume-76-number-6/programmed-privacy-promises-p3p-and-web-privacy-law/ (Consulted on 4 March 2019).

MIK, E., "Smart contracts: terminology, technical limitations and real world complexity", *Law, Innovation and Technology*, 2017, vol. 2, 269–300, https://www.ssrn.com/abstract=3038406 (Consulted on 14 February 2019).

O'HARA, K., "Smart contracts - Dumb idea", *IEEE Internet Computing*, 2017, vol. 2, 97–101, https://ieeexplore.ieee.org/document/7867719 (Consulted on 12 December 2019).

OMOHUNDRO, S., "Cryptocurrencies, smart contracts, and artificial intelligence", *AI matters*, 2014, vol. 2, 19–21, https://www.researchgate.net/publication/286243801_Cryptocurrencies_smart_contracts_and_artificial_intelligence (Consulted on 3 April 2019).

PARIS, C.E.C., "Contract Management: Design Parameters and Challenges to Implementation", *Scandinavian Studies in Law*, 2010, http://www.scandinavianlaw.se/pdf/49-10.pdf (Consulted on 27 February 2019).

PASQUALE, F.A., "A Rule of Persons, Not Machines: The Limits of Legal Automation", *George Washington Law Review*, 2018, https://www.ssrn.com/abstract=3135549 (Consulted on 10 March 2019).

PETERS, G.W. and PANAYI, E., "Understanding modern banking ledgers through blockchain technologies: Future of transaction processing and smart contracts on the internet of money" in *Banking Beyond Banks and Money*, Springer, 2016, 239–278, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2692487 (Consulted on 11 February 2019).

POON, J. and DRYJA, T., "The bitcoin lightning network: Scalable off-chain instant payments", 2016, https://www.bitcoinlightning.com/wp-content/uploads/2018/03/lightning-network-paper.pdf (Consulted on 1 May 2019).

POTTS, K., "Driving better fiscal management and revenue recognition with enterprise contract management technologies", *IACCM*, 15 December 2008, https://www.iaccm.com/resources/?id=8026 (Consulted on 27 February 2019).

RAMBERG, J., "INCOTERMS 2010", *Eur. JL Reform*, 2011, 380, https://heinonline.org/HOL/Page?handle=hein.journals/ejlr13&div=29&g_sent=1&casa_token=ozxVMlXCSJ0AAAAA:glq4ksO9vKgrVFAPz_AbgCiTpRR-OH_phXWbbqQyan0AiV0CCb3vv1WWQOX-E8Qom8vo_yrZ2A&collection=journals&t=1557489335 (Consulted on 10 May 2019).

RASKIN, M., "The Law and Legality of Smart Contracts", *Georgetown Law Technology Review*, 2017, vol. 2, 305–340, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2959166 (Consulted on 3 October 2018).

RICH, B., "How AI Is Changing Contracts", *Harvard Business Review*, 12 February 2018, https://hbr.org/2018/02/how-ai-is-changing-contracts (Consulted on 27 February 2019).

SAVELYEV, A., "Contract law 2.0: 'Smart'contracts as the beginning of the end of classic contract law", *Information & Communications Technology Law*, 2017, vol. 2, 116–134, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2885241 (Consulted on 3 March 2019).

SHI, N., "A new proof-of-work mechanism for bitcoin", *Financial Innovation*, 2016, vol. 1, 31, https://link.springer.com/article/10.1186/s40854-016-0045-6 (Consulted 5 April 2019).

SULIMAN, A., HUSAIN, Z., ABOUOUF, M., ALBLOOSHI, M. and SALAH, K., "Monetization of IoT data using smart contracts", *IET Networks*, 2018, https://www.researchgate.net/publication/326718966_Monetization_of_IoT_Data_using_Smart_Contracts (Consulted on 4 March 2019).

SURDEN, H., "Machine learning and law", *Washington Law Review*, 2014, 87, https://heinonline.org/HOL/Page?handle=hein.journals/washlr89&div=7&g_sent=1&casa_token=UMliB43s3moAAAAA:0XDk0lWORC1KDrPF-kYYBRB8zrr-FN45Xh1Kdb5G8x_o62AlJt13IEzQ5yxJHPZNm3_OaZwy (Consulted on 18 April 2019).

SURDEN, H., "Computable Contracts", *UCDL Rev.*, 2012, 629, https://lawreview.law.ucdavis.edu/issues/46/2/articles/46-2_surden.pdf (Consulted on 4 April 2019).

THORSRUD, E.S., *Long-term Bitcoin Scalability*, Thesis, Trondheim, Norwegian University of Science and Technology (NTNU), 2018, https://www.unit.no/vitenarkiv-i-bragekonsortiet (Consulted on 4 November 2018).

TJONG TJIN TAI, E., "Force Majeure and Excuses in Smart Contracts", *European Review of Private Law*, 2018, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3183637 (Consulted on 15 March 2019).

TJONG TJIN TAI, E., “Formalizing Contract Law for Smart Contracts”, *Tilburg Private Law Working Paper Series*, 2017, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3038800 (Consulted on 16 November 2018).

VAN DER ELST, C. and LAFARRE, A., “Bringing the AGM to the 21st Century: Blockchain and Smart Contracting Tech for Shareholder Involvement”, *European Corporate Governance Institute (ECGI) - Law Working Paper No. 412/2018*, 2018, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3219146. (Consulted on 22 February 2019)

VAN DER ELST, C. and VANDEVELDE, I., “De redelijke opzeggingstermijn bij het beëindigen van de verkoopconcessie van onbepaalde duur: een objectieve concessieformule”, *TPR*, 2014, vol. 4, 2487–2562.

WALDRON, J., “Vagueness in law and language: Some philosophical issues”, *Cal L. Rev.*, 1994, 509, https://heinonline.org/HOL/Page?handle=hein.journals/calr82&div=24&g_sent=1&casa_token=&collection=journals&t=1556982368 (Consulted on 4 May 2019).

WEBER, R.H., “‘Rose is a rose is a rose is a rose’–what about code and law?”, *Computer Law & Security Review*, 2018, https://www.sciencedirect.com/science/article/pii/S0267364918301882 (Consulted on 3 February 2019).

WERBACH, K.D., “Trust, but verify: Why the blockchain needs the law”, *Berkeley Technology Law Journal*, 2018, 488–550, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2844409 (Consulted on 5 March 2019).

WRIGHT, A. and DE FILIPPI, P., “Decentralized blockchain technology and the rise of lex cryptographia”, *SSRN*, 2015, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2580664 (Consulted on 4 May 2019).

WÜST, K. and GERVAIS, A., “Do you need a Blockchain?”, *IEEE*, 2018, 45–54, https://eprint.iacr.org/2017/375.pdf (Consulted on 22 November 2018).

YU, Y., *Contract as Code: Benefits and Challenges*, Ghent, University of Ghent, 2018, https://lib.ugent.be/fulltxt/RUG01/002/479/450/RUG01-002479450_2018_0001_AC.pdf (Consulted on 1 May 2019).

Newspaper articles

KAMINSKA, I., “Decentralised courts and blockchains”, *Financial Times*, 26 April 2016, https://ftalphaville.ft.com/2016/04/29/2160502/decentralised-courts-and-blockchains/ (Consulted on 4 May 2019).

MARR, B., "Here Are 10 Industries Blockchain Is Likely To Disrupt", *Forbes Magazine*, 16 July 2018, https://www.forbes.com/sites/bernardmarr/2018/07/16/here-are-10-industries-blockchain-is-likely-to-disrupt/#1d27a9c0b5a2 (Consulted on 24 December 2018).

NOONAN, L., "Banks find a use for blockchain: cross border payments", *Financial Times*, 12 November 2018, https://www.ft.com/content/57b1064a-c1a5-11e8-84cd-9e601db069b8 (Consulted on 24 December 2018).

Whitepapers, blog articles and various electronic sources

ARTIFICIAL LAWYER, "Allen & Overy Formally Adopts Kira Systems for AI Doc Review", *Artificial Lawyer*, 10 July 2018, https://www.artificiallawyer.com/2018/07/10/allen-overy-formally-adopts-kira-systems-for-ai-doc-review/ (Consulted on 2 March 2019).

BAJPAI, P., "Kodak, The Blockchain and Cryptocurrency: How Kodak Is Tapping Into Technology", *Nasdaq*, 28 January 2018, https://www.nasdaq.com/article/kodak-the-blockchain-and-cryptocurrency-how-kodak-is-tapping-into-technology-cm911406 (Consulted on 16 March 2019).

BROWN, R.G., CARLYLE, J., GRIGG, I. and HEARN, M., "Corda: An introduction", *R3 CEV*, 2016, https://docs.corda.net/_static/corda-introductory-whitepaper.pdf (Consulted on 3 January)

BUTERIN, V., "A next-generation smart contract and decentralized application platform", *white paper*, 2014, https://www.ethereum.org/ pdfs/EthereumWhitePaper.pdf/ (Consulted on 18 November 2018).

CANT, B., KHADIKAR, A., RUITER, A., BRONEBAKK, J., COUMAROS, J., BUVAT, J. and GUPTA, A., "Smart contracts in financial services: getting from hype to reality", *Capgemini Consulting*, 2016, 1–24, https://www.capgemini.com/consulting-de/wp-content/uploads/sites/32/2017/08/smart_contracts_paper_long_0.pdf (Consulted on 26 February 2019).

CLARK, B., "Blockchain and IP Law: A Match made in Crypto Heaven?", *Wipo Magazine*, 2018, https://www.wipo.int/wipo_magazine/en/2018/01/article_0005.html (Consulted on 16 March 2019).

DOURLENS, J., "Oracles: bringing data to the blockchain", *Ethereum Developers*, 9 October 2017, https://ethereumdev.io/oracles-getting-data-inside-blockchain/ (Consulted on 19 February 2019).

FRITZ, G. and TREICHL, L., "What's in a smart contract?", *Freshfields*, 2018, https://www.freshfields.com/en-gb/our-thinking/campaigns/digital/fintech/whats-in/whats-in-a-smart-contract/ (Consulted on 27 March 2019).

GRIGG, I., "On the intersection of Ricardian and Smart Contracts", 2017, http://iang.org/papers/intersection_ricardian_smart.html#ref_Szabo (Consulted on 26 February 2019).

GRIGG, I., "Financial cryptography in 7 layers", *Ian Grigg*, 2000, http://iang.org/papers/fc7.html (Consulted on 20 March 2019).

HAGAN, M., “6 Core Principles for Good Legal Design”, *Medium*, 7 November 2016, https://medium.com/legal-design-and-innovation/6-core-principles-for-good-legal-design-1cde6aba866 (Consulted on 4 March 2019).

ISDA and LINKLATERS, “Smart Contracts and Distributed Ledger – A Legal Perspective”, 2017, https://www.isda.org/a/6EKDE/smart-contracts-and-distributed-ledger-a-legal- perspective.pdf (Consulted on 20 January 2019).

KOK, A.S., “Write A Simple Contract On Top Of Ethereum”, *Coinmonks*, 2 May 2018, https://medium.com/coinmonks/write-a-simple-contract-on-top-of-ethereum-92b543594e84 (Consulted on 21 February 2019).

LIDDELL, P., SMART, M., RICHARDS, C. and GIBBY, A., “Supply chain capacity management – the key to value”, *KPMG Australia*, 21 March 2017, https://home.kpmg/au/en/home/insights/2017/03/supply-chain-capacity-management.html (Consulted on 27 February 2019).

MULLIGAN, C., SCOTT, J.Z., WARREN, S. and RANGASWAMI, J., “Blockchain beyond the hype; a practical framework for business leaders”, *white paper of the World Economic Forum*, 2018, http://www3.weforum.org/docs/48423_Whether_Blockchain_WP.pdf (Consulted on 4 March 2019).

NAKAMOTO, S., “Bitcoin: A peer-to-peer electronic cash system”, 2008, https://bitcoin.org/bitcoin.pdf (Consulted on 18 November 2018).

NG, J., “Escrow Service as a Smart Contract: The Business Logic”, *Medium*, 20 May 2018, https://medium.com/coinmonks/escrow-service-as-a-smart-contract-the-business-logic-5b678ebe1955 (Consulted on 25 March 2019).

ORCUTT, M., “States that are passing laws to govern ‘smart contracts’ have no idea what they’re doing”, *MIT Technology Review*, 29 March 2018, https://www.technologyreview.com/s/610718/states-that-are-passing-laws-to-govern-smart-contracts-have-no-idea-what-theyre-doing/ (Consulted on 24 March 2019).

QUALIUM, “Solidity – Ethereum’s Programming Language for Smart Contraction”, *Qualium Systems*, 18 July 2018, https://www.qualium-systems.com/blog/useful-it-articles-and-advices/solidity-ethereums-programming-language-for-smart-contraction/ (Consulted on 18 April 2019).

TOM, M., “The robots keep coming: DLA Piper makes major AI play with Kira software deal”, *Legal Business*, 14 June 2016, https://www.legalbusiness.co.uk/blogs/the-robots-keep-coming-dla-piper-makes-major-ai-play-with-kira-software-deal/ (Consulted on 2 March 2019).

STARK, J., “Making Sense of Blockchain Smart Contracts”, *Coindesk*, 4 June 2016, https://www.coindesk.com/making-sense-smart-contracts (Consulted on 1 June 2019).

SZABO, N., “Smart Contracts”, 1994, http://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/smart.contracts.html (Consulted on 24 April 2018).