PANEL 1

Is Arbitration Keeping Up with New Ways of Doing Business? The Arbitrators` Perspective

READINGS:

1. **Peter L. Michaelson and Sandra A. Jeskie, Arbitrating Disputes Involving Blockchains, Smart Contracts and Smart Legal Contracts**
2. **Ibrahim Mohamed Nour Shehata, Smart Contracts & International Arbitration**

Arbitrating Disputes Involving Blockchains, Smart Contracts and Smart Legal Contracts

Peter L. Michaelson, Esq. and Sandra A. Jeskie, Esq.*

Abstract

This paper addresses the use of arbitration to resolve disputes involving blockchain-based distributed ledgers (Blockchain Ledgers), Smart Contracts and Smart Legal Contracts. The first section of this paper addresses Blockchain Ledgers, Smart Contracts and Smart Legal Contracts, including how they work and the advantages these contract modalities provide over conventional oral/written contracts. The second section addresses various legal issues and the general nature of disputes that are likely to arise involving Blockchain Ledgers, Smart Contracts and Smart Legal Contracts and why arbitration is ideally suited to resolve these disputes. The last section addresses considerations for drafting arbitration clauses to be used with Smart Contracts and Smart Legal Contracts.

I. Introduction

Blockchain-based distributed (shared) ledgers (Blockchain Ledgers) provide an immutable, secure and tamper-evident alternative to conventional transactional modalities:[1] one which also yields enhanced accountability, traceability and transparency.

---

* Peter L. Michaelson is an Arbitrator, Mediator and Attorney with Michaelson ADR Chambers, LLC in New York, NY and Rumson, NJ. He arbitrates and mediates international and domestic disputes primarily involving IP, IT and technology, and secondarily other commercial areas. He is a panelist with various well-known and widely-respected ADR institutions, e.g., the AAA (including its commercial, large complex case, technology SEP-FRAND and other specialty panels) and its international division, the ICDR; WIPO; SIAC; HKIAC and CPR. He is a Fellow of the College of Commercial Arbitrators; a member of the National Academy of Distinguished Neutrals; and a Chartered Arbitrator and Fellow of the Chartered Institute of Arbitrators, and Chair Emeritus and Co-Founder of the New York Branch of CI Arb. He holds a LLM (Trade Regulation) from NYU School of Law, a JD from Duquesne University, and an MS in Electrical Engineering and a BS in Electrical Engineering and Economics both from Carnegie-Mellon University. Further information is available at www.plmadr.com. The author can be contacted at pete@plmadr.com. Sandra A. Jeskie is an arbitrator in complex disputes involving technology, intellectual property and complex commercial matters. She also serves the courts as a special master, mediator, and judge pro-tempore in a variety of complex business disputes. She holds an MBA in finance, a BA in computer science and before practicing law, she worked as a computer scientist. She serves as a neutral for AAA, ICDR and CPR, is a Fellow and Chair of the North American Branch of the Chartered Institute of Arbitrators (CIArb) and has been recognized by the Silicon Valley Arbitration & Mediation Center (SVAMC) as a leading technology arbitrator and mediator on their "Tech List". She is past President of the International Technology Law Association (ITechLaw) and a member of the American Law Institute (ALI). Further information is available at https://www.duanemorris.com/attorneys/sandraajeskie.html#tab_ADR.

[1] An early work dealing with a cryptographically secured chain of blocks, there to implement a system where document timestamps could not be tampered, was described in Stuart Haber et al, "How to time-stamp a digital document". *Journal of Cryptology,* Vol. 3, No. 2, January 1991, p. 99–111. In 1992, the system was expanded to allow several document certificates to be collected into one block. *David* Bayer et al," Improving the Efficiency and Reliability of Digital Time-Stamping", *Sequences,* Vol. 2. March 1992, p. 329–334. What appears to be the first conceptualization of blockchain was made by a person(s) known as Satoshi Nakamoto in 1998 -- though the exact identity of whom remains a mystery in the cryptographic field, when he published a paper describing the implementation behind the cryptocurrency BitCoin. Nakamoto incorporated hash methodology to timestamp blocks

The inherent benefits and hence growing adoption of Blockchain Ledgers, Smart Contracts and quite recently Smart Legal Contracts (the latter two being built on blockchains), across a wide range of the economy has caused and is now accelerating a fundamental paradigm shift that, in certain sectors of society, is increasingly displacing traditional written and oral contracts in favor computer-implemented, automatically executing blockchain-implemented agreements. For ease of reference and to prevent confusion, when Smart Contracts and Smart Legal Contracts are collectively discussed below, then, depending on context, they will be referred to as "smart agreements".

II. Background

A. Absolute Trust on the Blockchain

Trust is essential. All transactions are based on counterparties trusting each other. Parties will not transact with each other if they cannot establish sufficient trust in each other -- either directly or indirectly. Where counterparties either have either insufficient or no prior knowledge of each other and hence, little or no trust in each other, parties will traditionally employ an intermediary that each party trusts: whether it be an attorney, accountant, bank, underwriter, surety or other person or institution will depend on the specific nature of the transaction.

Blockchains establish impregnable trust: trust that cannot be violated, trust that is absolute -- and advantageously does so in an efficient, highly cost-effective and de-centralized manner. Blockchains eliminate the need to employ intermediaries. The following succinct historical perspective, quoted nearly verbatim, from the *MIT Technology Review* provides a rather instructive insight into why the need for trust drove the use of ledgers and double-entry accounting and ultimately blockchains:[2]

> "Beginning during the 14th century, Italian merchants and bankers, out of sheer necessity, developed and began using the double-entry bookkeeping method. This method, made possible by the adoption of Arabic numerals, gave merchants a more reliable recordkeeping tool, and it let bankers assume a powerful new role as middlemen in the international payments system. Yet it wasn't just the tool itself that made way for modern finance. It was how it was inserted into the culture of the day.
>
> In 1494 Luca Pacioli, a Franciscan friar and mathematician, codified their practices by publishing a manual on math and accounting that presented double-entry bookkeeping not only as a way to track accounts but as a moral obligation. The way Pacioli described it, for everything of value that merchants or bankers received, they had to give something back. Hence, the use of offsetting entries to record separate, balancing values—a debit

---

without requiring them to be signed by a trusted party and to reduce speed with which blocks are added to the chain. Satoshi Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System", 1998; https://bitcoin.org/bitcoin.pdf . Also see Arvind Narayanan, et al. *Bitcoin and cryptocurrency technologies: a comprehensive introduction.* Princeton University Press, 2016.

[2] Michael J. Casey and Paul Vigna, "In blockchain we trust", *MIT Technology Review*, April 9, 2018; https://www.technologyreview.com/s/610781/in-blockchain-we-trust/ .

> matched with a credit, an asset with a liability. Pacioli's morally upright accounting bestowed a form of religious benediction on these previously disparaged professions. Over the next several centuries, clean books came to be regarded as a sign of honesty and piety, clearing bankers to become payment intermediaries and speeding up the circulation of money. That funded the Renaissance and paved the way for the capitalist explosion that would change the world.
>
> Yet the system was not impervious to fraud. Bankers and other financial actors often breached their moral duty to keep honest books, and they still do —just ask Bernie Madoff's clients or Enron's shareholders. …
>
> A new form of bookkeeping might seem like a dull accomplishment. Yet for thousands of years, going back to Hammurabi's Babylon, ledgers have been the bedrock of civilization. That's because the exchanges of value on which society is founded require us to trust each other's claims about what we own, what we're owed, and what we owe. To achieve that trust, we need a common system for keeping track of our transactions, a system that gives definition and order to society itself. …
>
> The real promise of blockchain technology … is that it could drastically reduce the cost of trust by means of a radical, decentralized approach to accounting—and, by extension, create a new way to structure economic organizations.
>
> The benefits of this decentralized model emerge when weighed against the current economic system's cost of trust. … In 2007, Lehman Brothers reported record profits and revenue, all endorsed by its auditor, Ernst & Young. Nine months later, a nosedive in those same assets rendered the 158-year-old business bankrupt, triggering the biggest financial crisis in 80 years. Clearly, the valuations cited in the preceding years' books were way off. And we later learned that Lehman's ledger wasn't the only one with dubious data. Banks in the US and Europe paid out hundreds of billions of dollars in fines and settlements to cover losses caused by inflated balance sheets. … The crisis was an extreme example of the cost of trust. But we also find that cost ingrained in most other areas of the economy. Think of all the accountants … reconciling their company's ledgers with those of its business counterparts because neither party *trusts* the other's record. It is a time-consuming, expensive, yet necessary process.
>
> … [T]he internet of things, which it's hoped will have billions of interacting autonomous devices forging new efficiencies, won't be possible if gadget-to-gadget microtransactions require the prohibitively expensive intermediation of centrally controlled ledgers. …"

Ultimately, the ability to provide unassailable trust across a broad and growing spectrum of transactions drives the spread and adoption of Blockchain Ledgers. But a Blockchain Ledger by itself is one component. Smart contracts constitute software code that executes on the blockchain (i.e., on any of the computers that also hosts the Blockchain Ledger). This code, when executed, automatically processes applied external data (obtainable through, e.g., autonomous Internet-of-things (IoT) sensors) to yield corresponding entries on a Blockchain Ledger. What results is computer-implemented, automatically-executing agreements that do not

require any intermediary (whether human or institutional) at all, thus saving considerable cost and yielding considerable efficiency.

Mathematical rules and impregnable cryptography supplant trust previously reposed in fallible humans and institutions through traditional written and oral contracting and, through doing so, guarantee the integrity of the Blockchain Ledger. It is a version of what cryptographer Ian Grigg described as "triple-entry bookkeeping": one entry on the debit side, another for the credit, and a third into an immutable, undisputed, shared ledger.[3]

B. Legal Contracts, Smart Contracts and Smart Legal Contracts

1. Smart Contract

The Smart Contract Alliance,[4] an initiative of the Chamber of Digital Commerce[5], defines a Smart Contract as "computer code that, upon the occurrence of a specified condition or conditions, is capable of running automatically according to pre-specified functions. The code can be stored and processed on a distributed ledger and would write any resulting change into the distributed ledger." Smart Contracts can be used in various contexts, but they are particularly useful when integrated into Blockchain Ledgers. As the use and development of distributed ledger technology has dramatically increased, considerable confusion had arisen regarding the differences between Smart Contracts and conventional (non-computer implemented) legal contracts.[6]

A fundamental difference between a Smart Contract and a legal contract is the authority that dictates enforcement of the contract: essentially, a Smart Contract automatically enforces a relationship specified in code (the computer software that, when executed, implements the Smart Contract); whereas, a judicial system, arbitrator or some other authority enforces the terms of a legal contract.[7] A Smart Contract contains no independent means of enforcement. It is simply executed when a predefined condition, determined by a sensor or a so-called "oracle"[8], either occurs or, within a specified period of time or under some other constraint, does not occur. Many aspects of legal contracts, such as those which rely on the exercise of human judgment and insight, are presently incapable, and may never be capable, of being represented by condition-based functions used in Smart Contracts.

2. Smart Legal Contract

A Smart Legal Contract is considerably more sophisticated and complex than a Smart Contract. The former, having both "smart" (computer-executed) and "non-smart" (traditional text-based) clauses, is amalgam of a Smart Contract and a legal contract. The Smart Contract Alliance

---

[3] *Ibid.*
[4] https://digitalchamber.org/initiatives/smart-contracts-alliance/,
[5] https://digitalchamber.org/
[6] Mark M. Higgins, "Blockchain in Energy: Smart Legal Contracts on the Rise", *National Law Review*, July 26, 2019; https://www.natlawreview.com/article/blockchain-energy-smart-legal-contracts-rise .
[7] *Ibid*.
[8] Oracles retrieve and verify external data for blockchains and smart contracts.

defines a Smart Legal Contract as "a Smart Contract that articulates and is capable of self-executing, on a legally-enforceable basis, the terms of an agreement between two or more parties."[9] For example, a Smart Legal Contract may include a smart payment clause, with code determining the amount due for a particular payment and, based on monitoring a payee's bank account, whether that payment was made by a date certain or not, while all of the other provisions of the contract (Definitions, Jurisdiction clause, Force Majeure clause, ...) appear solely in regular natural language text.

In that regard, the Accord Project, a non-profit open-source consortium aimed at transforming contract management and contract automation, is developing an open, standardized format for Smart Legal Contracts [10] along with a software ecosystem and open-source tools to digitize new or existing legal contracts, connect them to web services and deploy them to the cloud or a blockchain platform.[11] The Accord Project views a Smart Legal Contract as both a human- and machine-readable agreement that is digital, consisting of natural language and computable components. The human-readable aspect of the document ensures that signatories, lawyers, contracting parties and others are able to understand the contract. The machine-readable aspect enables the contract to be interpreted and executed by computers, making the document "smart". Its goal is that anyone, through use of those tools and the ecosystem, can draft Smart Legal Contracts in a standardized neutral, technology agnostic format, once and then use and reuse it, as often as desired, across a variety of supported technologies.[12]

The Global Legal Blockchain Consortium (GLBC) is another non-profit organization that is highly active in this area. The GLBC aims to drive the adoption and standardization of using blockchain technology throughout the legal industry while ensuring data integrity, authenticity and privacy and improving the security and interoperability of the global legal technology ecosystem. The GLBC comprises over 300 large companies, law firms, software companies and universities, all seeking to collaboratively develop standards governing the use of open-source blockchain technology in the legal industry.[13] In 2019, the American arbitration Association (AAA) executed a memorandum of understanding with the GLBC. In 2020, the AAA plans to spearhead establishment of a GLBC-sponsored alternate dispute resolution community of interest to explore "on-chain" and "off-chain" arbitration of blockchain disputes.

3. Ricardian Contract

---

[9] *Ibid*.

[10] https://www.accordproject.org/. Clyde & Co (a London-based global law firm specializing in insurance and international trade) developed an off-the-shelf connected parametric insurance contract for use by insurers through its Smart Contract group, Clyde Code. The contract has been built in collaboration with Smart Legal Contracts platform Clause and according to the specifications developed by The Accord Project, although it can be deployed on other systems and platforms. "Clyde & Co launches connected parametric insurance contract", *Clyde & Co. Newsletter*, May 15, 2019. In the US, Latham & Watkins has teamed up with ConsenSys to develop a Smart Legal Contract that automates convertible note agreements. This effort, like other efforts to create legally enforceable code, necessitates the engagement of an attorney. Counsel is necessary to determine the parameters of a specific deal and move beyond a standard suite of documents. *Higgins, cited previously*.

[11] https://docs.accordproject.org/

[12] https://www.accordproject.org/

[13] https://legalconsortium.org/what-is-the-glbc/.

The Ricardian contract, conceived of by financial cryptographer, Ian Grigg, is a contract represented in plain text and in digital code, digitally signed to provide it with all the elements of a standard legal contract.[14] Grigg defined the role of the Ricardian contract as a document that attempts to recognize the intent of the agreement between the parties, while the smart contract is the machine that executes that agreement.[15] *Forbes* described the Ricardian Contract as a smarter and more useful digital contract.[16]

There are obvious efficiency and cost advantages to Smart Legal Contracts and Ricardian contracts. Not surprisingly, various parties in the legal industry have started to capitalize on implementing and using these contracts, though these efforts, as with the Accord Project, are still rather early in the development phase.[17]

C. Illustrative Smart Contract Examples

As the benefits of using Blockchain Ledgers and smart agreements are increasingly recognized in practice, applications of these technologies, which are likely to only exponentially increase with time, are being envisioned across many diverse facets of commerce, industry and government. The following examples clearly reflect the breadth of these applications and the societal benefits obtainable through these technologies.

1. Securing the U.S. Electrical Grid

During a frigid day in December 2015, the Ukrainian power grid was hacked with more than 230,000 Ukrainians then losing power for an afternoon. The hackers exploited a software vulnerability in a central control system to attack Ukrainian power plants. In the U.S., power plants are fed data from the Supervisor Control and Data Acquisition (SCADA) system that American power plants use to decide how power to generate and where to send it. As SCADA can be a huge central point of attack, the U.S. Dept of Energy recently awarded a $400,000 grant to researchers at Carnegie-Mellon University to substantially harden SCADA from hacking by placing incoming data on a Blockchain Ledger. By doing so, an attacker would need to successfully hack not one, but tens or hundreds of computers depending on the number of nodes in the blockchain -- which is an extremely difficult task.[18]

2. Providing Safety in the U.S. Food Supply Chain; Locating Sources of Counterfeit Goods

Blockchain Ledgers can be used to secure food supply chains by allowing users to quickly trace the origin and provenance of contaminated foodstuff back to its source. Within the past few

---

[14] "Filling in the Missing Piece of Smart Contracts," Nasdaq, August 15, 2018; https://www.nasdaq.com/articles/filling-missing-piece-smart-contracts-2018-08-15

[15] *Id.* (citing Ian Grigg, "On the intersection of Ricardian and Smart Contracts", February 2015).

[16] Chao Cheng-Shorland, "Moving Beyond Smart Contracts: What Are The Next Generations Of Blockchain Use Cases?", December 5, 2018; https://www.forbes.com/sites/forbestechcouncil/2018/12/05/moving-beyond-smart-contracts-what-are-the-next-generations-of-blockchain-use-cases/#19bb06ad13e5

[17] https://docs.accordproject.org/docs/accordproject.html

[18] "Securing the Energy Grid with Blockchains", *Carnegie-Mellon Engineering Magazine, Carnegie Institute of Technology*, Fall 2019, p. 28.

years, a number of multi-state instances of e-coli contamination, which caused illness among a small number of consumers and in some rare instances death, has been found in agricultural products, such as romaine lettuce, originating from various growers, agriculturally-related facilities or growing regions in California and other producing states. Historically, the Centers for Disease Control required considerable time and effort to manually trace contaminated produce from the affected consumers outward and ultimately locate the source of contamination to a specific producers, facilities or regions for appropriate remediation.[19] To appreciably shorten this time, each and every different point along a chain of custody starting with an individual grower, through all intermediate points where possession changes, to ultimately an endpoint in the chain which either uses or sells the produce to a consumer can be permanently recorded, via Smart Contracts, on a Blockchain Ledger. The ledger provides an irrefutable shared record of ownership, location and movement along every facet of the food supply chain, thus increasing efficiency, transparency and trust, with information being simultaneously and securely available to each entity along the chain as well as regulators. [20] By simply inspecting the ledger, a regulator can pinpoint, within seconds rather than weeks, a particular grower, facility or region for investigation, thus dramatically reducing the spread of contamination and the number of instances of consumer illness, thus significantly improving public safety.

Similarly, Blockchain Ledgers can be used to find the source of counterfeit or faulty goods by tracing the origin and provenance of previously shipped goods, including, e.g., investigating industry certifications, tracking restricted or dangerous components and discovering storage anomalies.[21]

For example, in June 2019, the FDA chose Merck & Co, IBM, KPMG, and Walmart to form a pilot project aimed at evaluating the use of blockchain to protect pharmaceutical product integrity, by identifying and tracing certain prescription drugs as they were distributed within the US. The project was authorized under the US Drug Supply Chain Security Act, an act which increased the FDA's ability to help protect consumers from exposure to counterfeit, stolen, contaminated or otherwise harmful drugs.[22]

III. The Technologies

After establishing the underlying need for Blockchain Ledgers and some of the advantages of those ledgers, Smart Contracts and Smart Legal Contracts, we will now discuss how they work.

---

[19] https://www.cdc.gov/ecoli/2018/o157h7-11-18/index.html

[20] https://www.ibm.com/downloads/cas/1VBZEPYL and https://www.ibm.com/blockchain/industries/supply-chain . Also, Sloane Brakeville et al, "Blockchain basics: Glossary and use cases", *IBM Developer*, August 21, 2017; https://developer.ibm.com/tutorials/cl-blockchain-basics-glossary-bluemix-trs/ .

[21] https://www.ibm.com/downloads/cas/1VBZEPYL. Also see "National Action Plan for Blockchain -- The Need for a Comprehensive, Coordinated, Pro-Growth Approach to Developing Blockchain Technology in the United States", *Chamber of Digital Commerce*, February 2019; https://digitalchamber.org/wp-content/uploads/dlm_uploads/2019/02/National-Action-Plan-for-Blockchain1.pdf .

[22] https://www.worldipreview.com/news/us-border-agency-tests-ip-blockchain-solution-19392?utm_source=2.+World+IP+Review&utm_campaign=f765573797-WIPR_Digital_Newsletter_30012020_COPY_01&utm_medium=email&utm_term=0_d76dcadc01-f765573797-27049533.

A. A primer on the technologies

1. Blockchains and Blockchain Ledgers

A blockchain stores transaction data in blocks. A typical such block (labelled Block n) is depicted in Figure 1 below.

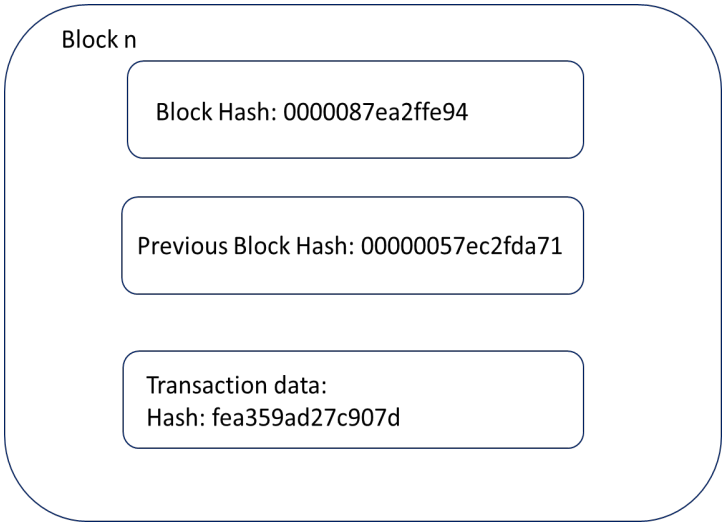


Figure 1 -- Typical Blockchain block

As shown, the block contains transaction data (as the specifics of which are irrelevant to this explanation, they have been omitted for simplicity) for a given transaction and its hash value. Transactions can represent almost anything (often referred to as a "digital asset"), such as actual exchanges of money, as occurs on blockchains that underlie cryptocurrencies like Bitcoin. Alternatively, transactions could represent exchanges of other assets represented digitally, such as digital stock certificates, deeds, bills of sale, transfers and so forth. For any given transaction, its transaction data contains valid pertinent information specifying the nature of the underlying transaction, such as, e.g., the specific goods or amount of money involved, the parties involved and their locations; and also a timestamp of when (date and time) that transaction occurred. That data is collectively processed through a cryptographic HASH function, which is a predefined mathematical algorithm (e.g., the SHA256 algorithm[23]) that yields a hash value. The moment a block is created, its host computer automatically computes and includes its block hash value. The hash algorithm has critical properties essential to cryptography and here blockchains: the algorithm is irreversible meaning that the underlying input information cannot be determined from its hash value; the algorithm is deterministic meaning that the same input data will always generate the same hash value; the hash value can be computed relatively quickly; and, importantly, a small change in the input data will so extensively change the resulting hash value that the new hash value appears to be uncorrelated (i.e., random) with respect to the immediately preceding hash value. The block also contains the hash value for the entire block, i.e., the block hash, and the block hash for an immediately preceding block in the blockchain. The block hash results from applying the HASH function to the hashed transaction data and the previous block

[23] *See., e.g.*, https://xorbin.com/tools/sha256-hash-calculator .

hash, hence effectively creating a hash of a hash (the result of this operation is commonly referred to, in the cryptography field, as a "Merkle Root").[24]

The existence of the prior block hash value in each block is what allows the blocks to be linked, i.e. chained, together. This is shown in Figure 2 which depicts three successive blocks in the blockchain, Blocks n-1, n and n+1. Each block stores information for a corresponding transaction. As the number of transactions grows, so does the number of blocks in the blockchain and hence its size.

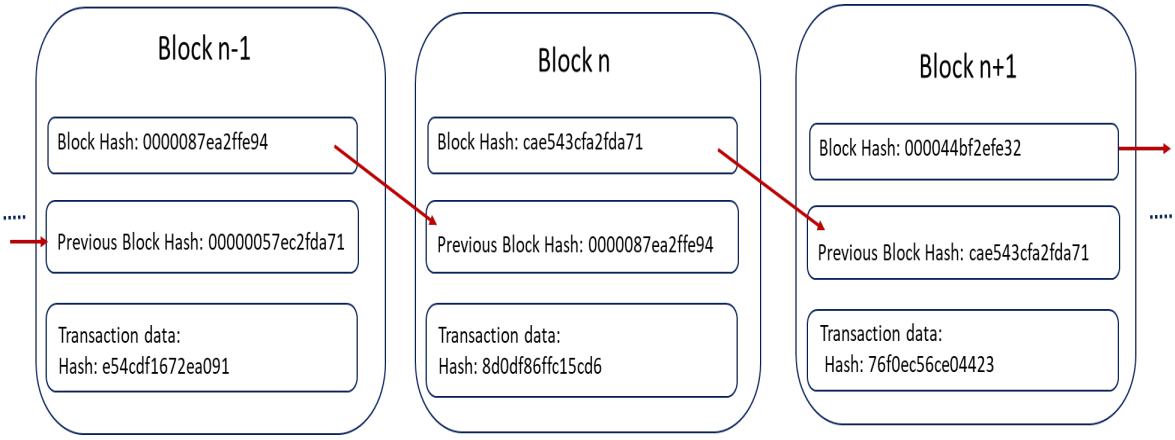


Figure 2 -- Interconnected Blockchain blocks

All the transaction data stored across all the blocks in a blockchain collectively forms a ledger.

Conventional business networks, simplistically illustrated by that depicted in Figure 3 below, for recording transactions rely on each party, A-D, to write transaction data into its own database (containing respective Ledgers A-D) and communicating transaction and other data through a data network, such as the Internet, with every other party making corresponding updates to their own ledgers. This arrangement requires all four parties to maintain four separate ledgers. Critically, this arrangement is susceptible to being compromised because if any one ledger is improperly altered due to fraud, cyberattack or just a simple human mistake, incorrect transaction data will propagate to and adversely affect transaction data stored in all the other ledgers.

[24] Manav Gupta, "Blockchain for dummies, IBM Limited Edition", *IBM Corp.* (© 2017, John Wiley & Sons), p. 13-14; http://gunkelweb.com/coms465/texts/ibm_blockchain.pdf . Also Anastasiia Lastovetska, "Blockchain Architecture Basics: Components, Structure, Benefits & Creation,", MLSDev, January 31, 2019; https://medium.com/@MLSDevCom .

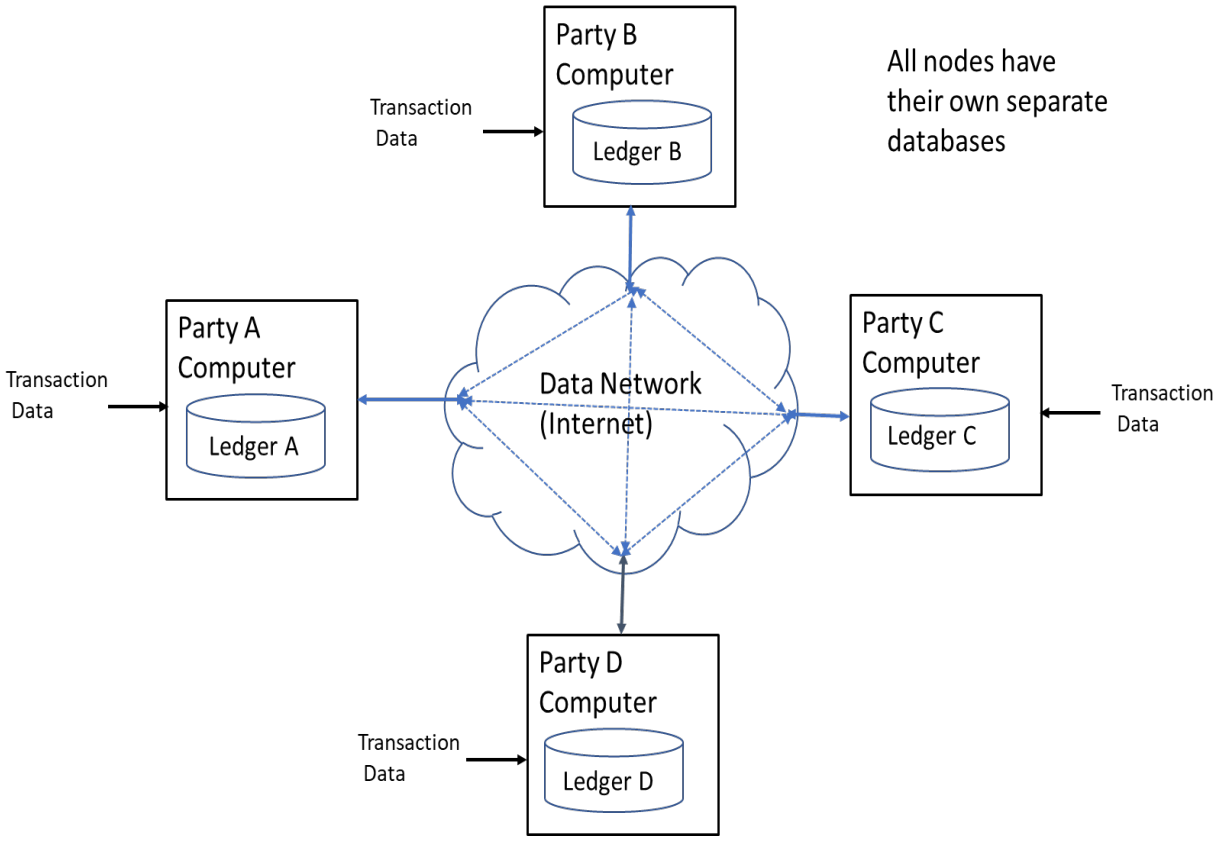


Figure 3 -- Conventional Business Network

By contrast, Figure 4 depicts a blockchain network. For ease of understanding, it is a simple four-node network consistent with that shown in Figure 3, though in actuality, blockchain networks can contain tens, hundreds or thousands of "nodes" (such as that used in a public blockchain for Bitcoin and other cryptocurrencies). The blockchain, as shown in Figure 4, is stored in multiple copies across multiple independent computers, each forming a node in the data network, with each node storing a complete local copy of the blockchain, hence forming a decentralized structure. As the transaction data stored within the blockchain on each node constitutes a complete copy of the ledger, by virtue of the blockchain being copied across all nodes, the ledger is effectively distributed, in copies, across all the nodes.[25] As will be described in further detail below, each node writes all transactions, once validated, into its replica of the blockchain, thus the common ledger is always synchronized across all four notes. Each node can be a PC, workstation, server, laptop, mobile device or any computer-based device that has network connectivity and sufficient processing power to execute software application programs which implement the blockchain and related functionalities. Further, although each node is illustrated as a physical element located outside the data network, that node can just as easily be located within a cloud environment and implemented as either physical or, more likely, virtualized. Various vendors, including Microsoft and IBM, currently offer so-called "Blockchain-as-a-Service" through which the vendor will design and implement, in its respective

[25] Michael J. Casey and Paul Vigna, "In blockchain we trust", *MIT Technology Review*, April 9, 2018; https://www.technologyreview.com/s/610781/in-blockchain-we-trust/ .

cloud environment (Microsoft Azure and IBM Cloud), an entire virtualized blockchain infrastructure (Microsoft Azure Blockchain and IBM Blockchain Platform) based on a customer's need with pay-as-you-go, fee-for-use based pricing (i.e. utility type pricing), thereby freeing the customer of the considerable effort and cost of designing and implementing its own blockchain distributed ledger system.[26]

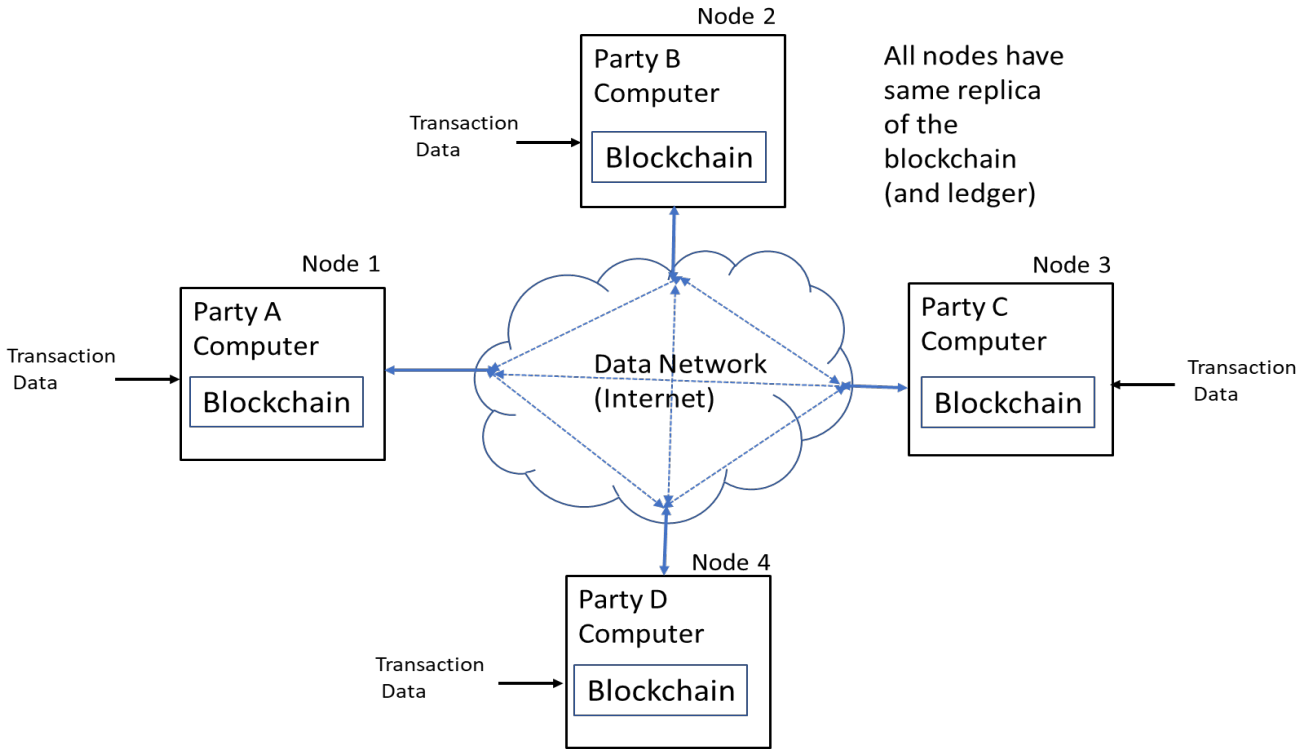


Figure 4 -- Four-node Blockchain Network

No single entity controls the ledger. Any node can make a change to the ledger by requesting that a new block be added to an end of the blockchain. Once that request is made, the requesting node sends the request and the new block to every other node on the network. Each node, that receives the new block, verifies that block and determines whether its transaction data is valid. The new block will only be added if pre-defined rules implemented through a consensus protocol are satisfied. That protocol is a mathematical algorithm which requires at least a majority (and sometimes all, depending on the amount of consensus to which the blockchain is configured) of the nodes which received the new block to agree with the change. Once consensus is reached and communicated to all the nodes on the network, all those nodes will simultaneously update their copies of the ledger by inserting the new block. If any node attempts to add a block to the ledger without achieving consensus, all the other nodes automatically reject the attempt as invalid and the addition is not made. Once a block is added to the blockchain, the entry is permanent. It cannot be deleted. It cannot be altered. Blocks are entered in an append-only fashion; they are only added to the end of the blockchain: one after another. Should a node

[26] For further information on BasS offerings from Microsoft, see https://azure.microsoft.com/en-us/solutions/blockchain/ ; and for IBM, see https://www.ibm.com/blockchain .

subsequently request a modification to an existing block, such as in the case of a transaction that has been modified (as to amount, such as a refund or discount, change of a party or location), that node requests the addition of a new block which provides the modification. No existing block is modified. As a result, the blockchain records, stores and reflects each and every action that involved it thus forming a complete sequential historical ledger of transactions.

A blockchain network has the following key characteristics:

i. Consensus -- For a transaction to be valid, at least a majority (and in some instances all) of the parties (participants) on the blockchain must agree on its validity.

ii. Provenance -- By virtue of each and every transaction affecting a digital asset being entered into the blockchain, all the participants know where that asset originated and how its ownership changed over time.

iii. Immutability -- No participant can tamper with a transaction after it has been entered into the Blockchain Ledger. If a transaction is in error, a new transaction must be entered to reverse the error and both transactions are visible on the blockchain.[27]

The need to achieve consensus among replicated blockchain nodes coupled with the linkage of successive blocks in each replica through their block hash values renders a blockchain, for all practical purposes, impervious to hacking.

Before a node can add a new block to the blockchain, it must first achieve consensus based on responses from other nodes as to the validity of that new block. If that new block is not valid, it will not be accepted and added to the blockchain, thus thwarting any attempt to illicitly change a single block.

In order for a hacker to successfully change a particular transaction on the blockchain, that hacker would not only need to change the corresponding block containing that transaction on any one node but also, due to the distributed nature of the ledger, the same block on each and every other node of the chain. Further, since each block contains its own block hash value and that of its immediately prior block, the hacker would also need to properly change the hash value on each and every block in the blockchain subsequent to the corresponding block and on each replica of the blockchain stored on each and every node. All of this, practically speaking, is a virtually impossible task. Thus, a Blockchain Ledger provides its users with impregnable trust: they need not trust each other, but each can repose undeniable trust in the distributed ledger itself.

Within this general framework, many differences can arise depending on specific characteristics of the blockchain network. For example, public "permissionless" blockchain networks exist through which any computer can become part of the network -- as is the case with cryptocurrencies such as Bitcoin; and private "permissioned" ledgers to which access is strictly limited to certain credentialed users having appropriate "permissions" and, for those users, certain purposes. Permissioned ledgers are typically used by a particular group of organizations

[27] Gupta, *cited infra*, p. 15.

(parties) that are transacting together, such as a supply chain, which require a common, secure, immutable record-keeping system but are otherwise independent of each other and may well not fully trust each other.[28]

A principal implementational difference between permissioned and permissionless ledgers is the inclusion in the latter of an additional verification process as part of determining consensus, which in the context of cryptocurrencies, such as Bitcoin, is called a "mining" step. Through that step, a node, which requests a new block be added to the blockchain, calculates a so-called "proof of work" (which consumes a huge amount of processing power to complete) in order to validate the new block.[29] As permissioned ledgers are the norm in commercial blockchain applications, this paper will solely focus on those ledgers.

Further, there are different consensus algorithms that can be used in a Blockchain Ledger along with significant variations in the number of nodes that are required to determine and communicate consensus, the details of all of which are well beyond the scope of this paper and hence will not be discussed.

Figures 5-7 diagrammatically and successively depict, in a simplified fashion, messaging and corresponding operations that occur within a blockchain network whenever a new block is being appended to the blockchain. To facilitate understanding, these figures use the same four-node blockchain network shown in Figure 4. For further simplification, this example assumes that the consensus algorithm is implemented only within one node and requires complete consensus, i.e. every node must validate a new block before it can be added to end of the blockchain.

[28] Loic Lesavre et al, "A Taxonomic Approach to Understanding Emerging Blockchain Identity Management Systems", *NIST (National Institute of Standards and Technology) Cybersecurity White Paper (Draft)*, July 9, 2019; https://doi.org/10.6028/NIST.CSWP.07092019-draft . Also, Gupta, *cited infra*, p. 16.
[29] *See, e.g.*, https://cointelegraph.com/explained/proof-of-work-explained .

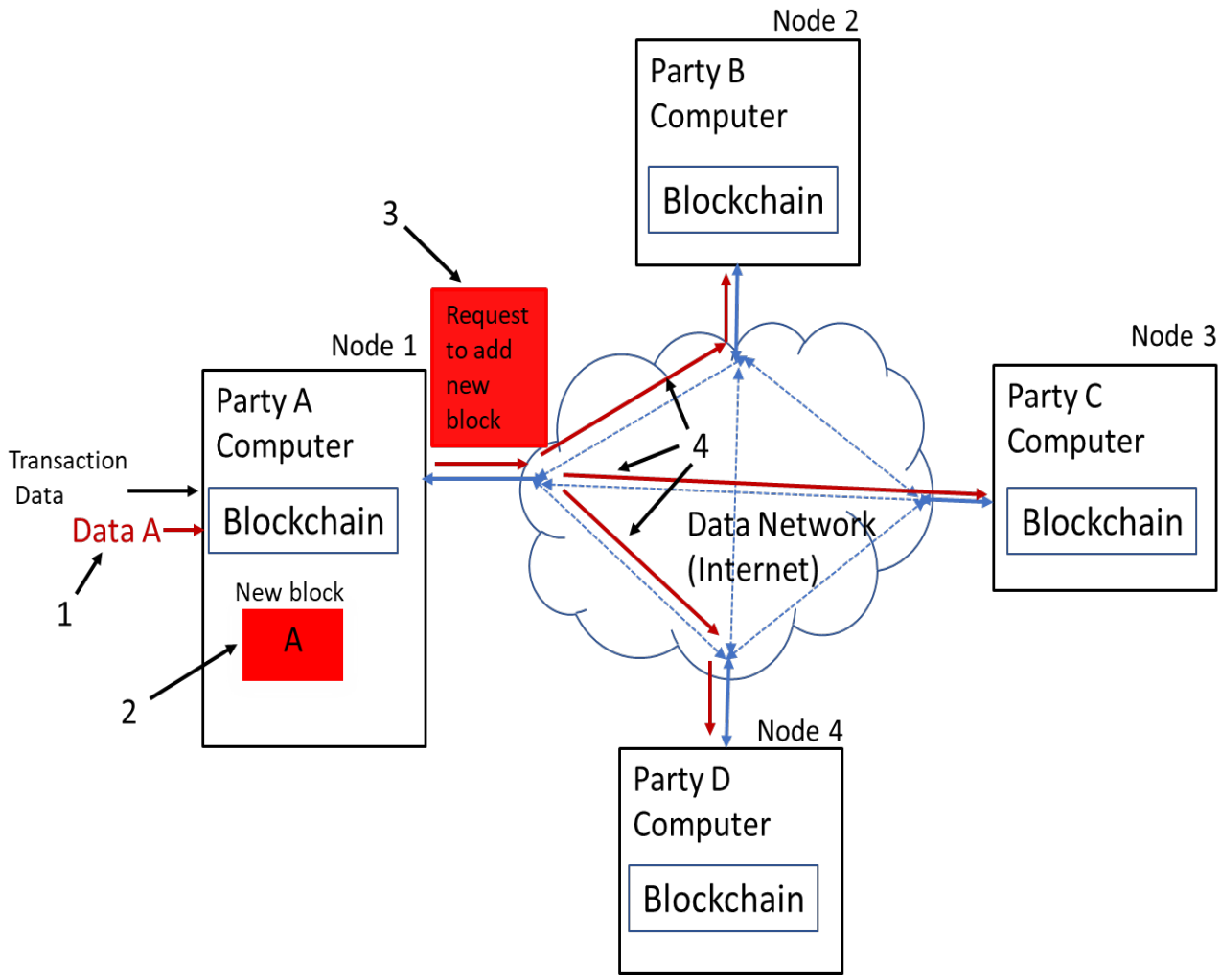


Figure 5 -- New Block Generation

As illustrated in Figure 5, a new transaction occurred resulting in Data A being sent to Node 1; the operation symbolized by numeral 1. In response, Node 1 constructs, as symbolized by numeral 2, a new block containing this data and Request 3 to add that block to the blockchain. Node 1 then transmits, as symbolized by numeral 4, Request 3 to each of the other nodes.

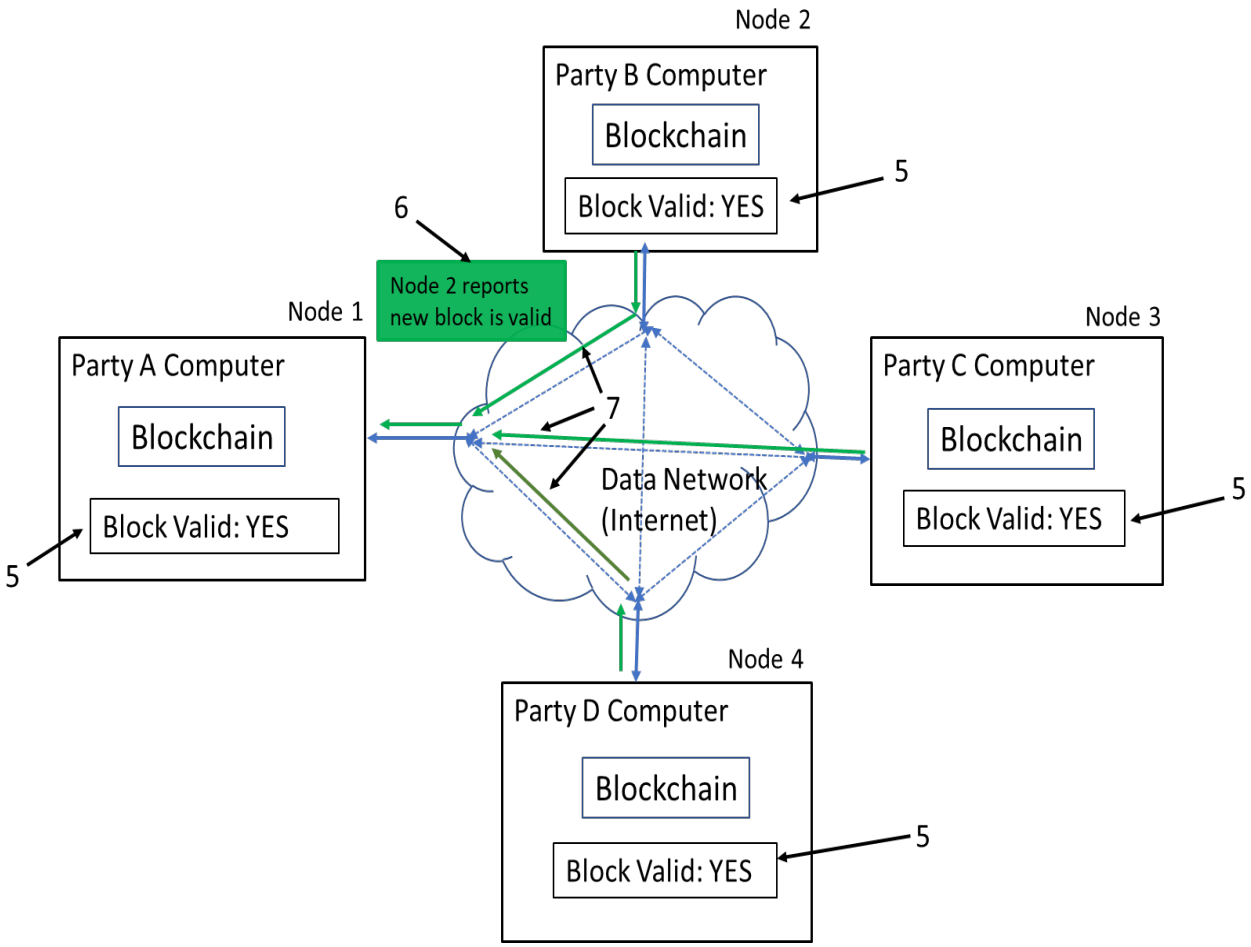


Figure 6 -- Validity Determination

Next, as shown in Figure 6, each node independently determines whether the new block is valid with, this operation symbolized by block 5. Each block then transmits a message, symbolized by message 6 from Node 2, providing its results back, as symbolized by lines 7, to the requesting node, Node 1. Thereafter, as shown in Figure 7 below, Node 1 determines, as represented by block 8, whether consensus exists that the new block is valid, i.e. whether all the blocks agree. If, as here consensus exists, then Node 1 generates Add New Block command 9 and then transmits that command to each of the other nodes, the latter operation symbolized by lines 10. Each node, in response to the command then actually appends the new block onto the blockchain replica stored within that node as the last block, that being symbolized by block 11. Alternatively, each node can broadcast its validity message throughout the network with every node then making its own consensus determination, based on its own validity result determination and all the validity messages it receives, and in response merely adding the new block or not to its own blockchain replica without sending a command to each of the other nodes instructing any of the latter to do so.

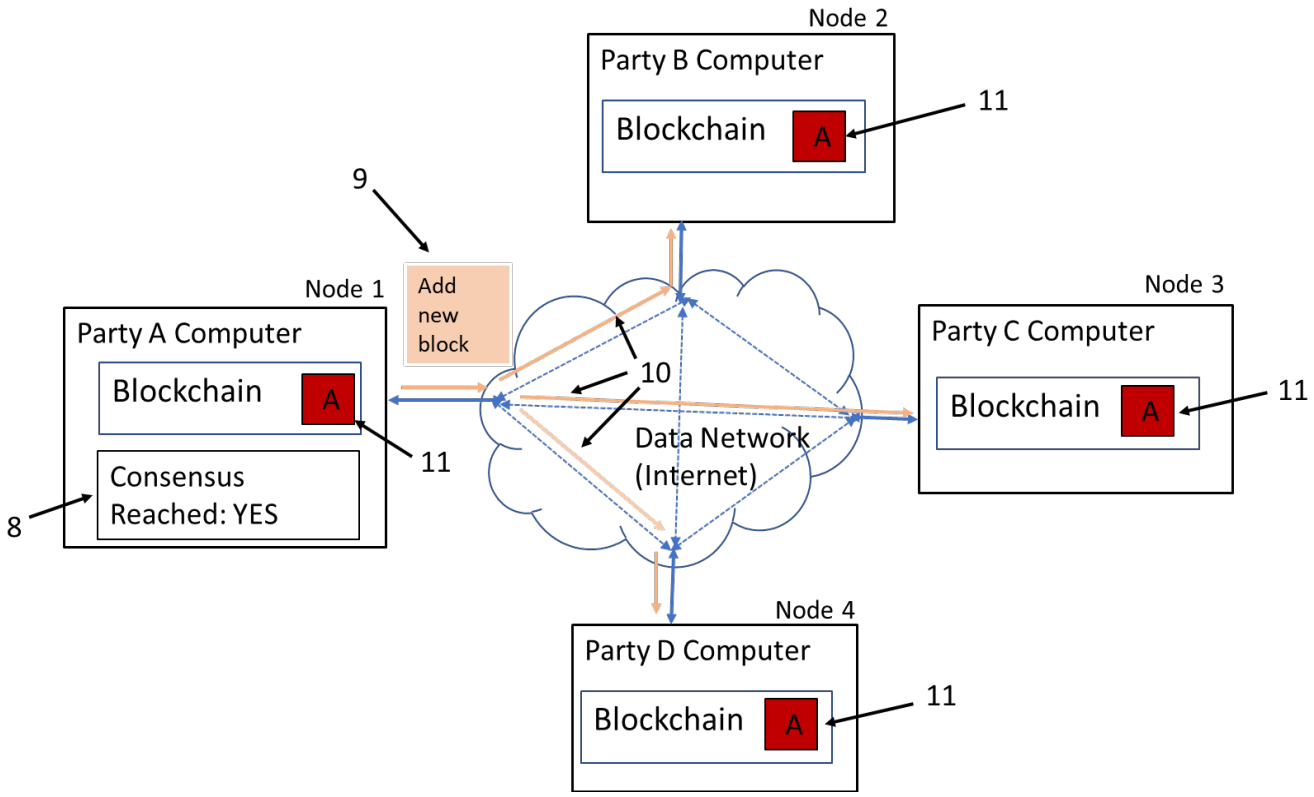


Figure 7 -- Appending New Block to Blockchain

As the reader can now readily appreciate, Blockchain Ledgers, due to the inherent replication of the entire blockchain across all nodes in the network and the requirement that all nodes perform all the same tasks (with some exceptions regarding which nodes determine consensus), are highly redundant and thus exceedingly inefficient both in terms of storage and processing. Yet, that redundancy is just what enables, in practice, Blockchain Ledgers to provide an immutable degree of trust --- one that cannot be compromised or violated -- to all its participants that any transaction recorded in the ledger has not be illicitly modified, altered or changed in any way.[30]

2. Smart Contracts and Smart Legal Contracts

Figure 8 below depicts, at a very high-level fashion, the additional components within a Blockchain network node for implementing Smart Contracts and Smart Legal Contracts, as shown, respectively, in the block diagrams on the left and right sides of the figure.

[30] Demiro Massessi, "Blockchain Consensus And Fault Tolerance In A Nutshell", *Coinmonks*, January 6, 2019; https://medium.com/coinmonks/blockchain-consensus-and-fault-tolerance-in-a-nutshell-765de83b8d03.

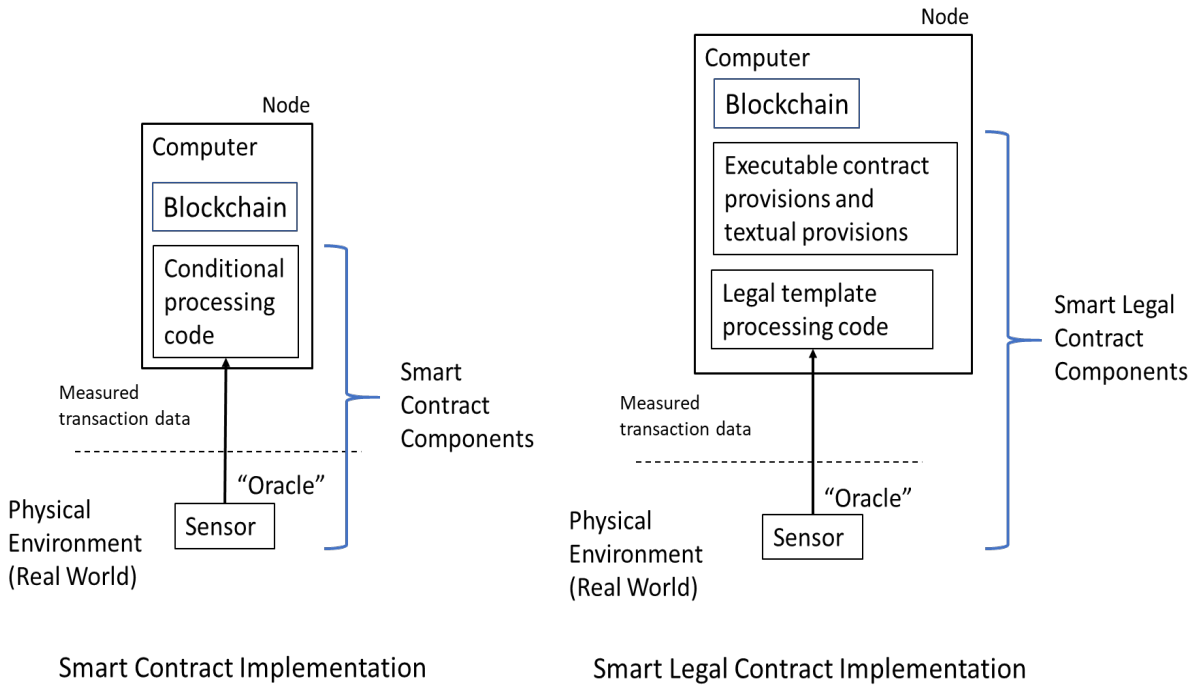


Figure 8 -- Smart Contract and Smart Legal Contract Implementation

As previously discussed, Smart Contracts are self-executing computer code programmed to execute transactions when pre-defined conditions occur, i.e. they automatically enforce a relationship specified in code. As Smart Contracts run on the blockchain, they run exactly as programmed without, in practice, any possibility of censorship, downtime, fraud or third-party interference. The contract code and conditions are publicly available on the Blockchain Ledger.[31] That code, basically implementing conditional logic, accepts measurements, in the form of measured real-world transaction data, whether from a remote sensor or from some other source which, with appropriate data retrieval and verification functionality, can also, as shown, be an oracle. The sensor measures some aspect of the real world. The code implements specific and alternate contract terms and is triggered depending on the value of the incoming data, whether measured by the sensor or directly applied through a remote source. The data may simply reflect, in a binary "YES/NO" manner, whether a given event has occurred or not. The logic is typically implemented using "if then else" type conditional processing, namely if the data value equals X, then perform step A, else perform step B.

A Smart Contract is not synonymous with a legally binding contract. Smart Contracts can be and are being used in applications that have very little, if anything, to do with acting as a legally binding contract (e.g., supply-chain management, self-sovereign identity, and provenance tracking). That said, Smart Contracts can constitute elements of a legal binding contract under common law.[32]

[31] https://blockgeeks.com/guides/ethereum/.
[32] "Smart Contrasts: Is the Law Ready?". Smart Contracts Alliance, Chamber of Digital Commerce, September 27, 2018, p. 23; https://digitalchamber.org/smart-contracts-whitepaper/.

For example, under a Smart Contract, payment for goods is due a seller when certain goods are delivered to a buyer. At the buyer's facility, an employee at a loading dock may use a handheld barcode reader to scan barcoded information printed on shipment documents for all incoming shipments to confirm receipt. The sensor in this instance is the barcode reader. Once the scanned data is received by the computer, that triggers the Smart Contract logic which, in turn, instructs payment to be made to the Seller and a new block added to a Blockchain Ledger reflecting that event. If the goods have not arrived by a predefined date, then the logic may invoke an alternate action, such as notifying the seller of non-delivery and instruct the Blockchain Ledger to add a new block reflecting that event. The Smart Contract here is simply the sensing of the occurrence or non-occurrence of a delivery.

A Smart Legal Contract, being far more sophisticated than a Smart Contract, is implemented with both computer-executed contractual clauses and traditional text-based clauses. As discussed, a Smart Legal Contract, pursuant to, e.g., the framework promulgated by the Accord Project, relies on using a legal template and accompanying executable code. When the executable code is processed, real-time sensed measurement data is inserted into the coded template and, based on the value of the data and the instructions set forth in the template code, specific contractual action as specified in the template is then automatically invoked and an accompanying new block, reflecting that action, is established and added to the blockchain.[33]

For example, a Smart Legal Contract may contain a smart payment clause using executable code to determine a specific amount due a domestic supplier in an international sales transaction, then invoke its payment and finally, upon the supplier's receipt of that amount, write a new block reflecting that transaction into a Blockchain Ledger.

Specifically, a computer is informed, via a message generated by the sensor, that the supplier has performed its contractual obligation (delivery of purchased goods or rendering of a purchased service) for the customer. The message includes, e.g., the names and addresses of the parties, the sale price to be paid in a domestic currency (e.g., US dollars), the goods/service furnished, and the parties' respective banking information. In response, the code, during its execution, ascertains, based on address data of the parties, whether the payment is to be debited from the customer's bank account in a particular foreign currency (e.g., Euros) and, if so, determines the applicable foreign exchange rate for the transaction. The executing code then calculates the amount of the payment due in the foreign currency based on that rate, adds in any applicable currency translation charge, and automatically instructs the customer's bank to debit that full amount from the customer's account. The code also instructs the customer's bank to send that amount to the supplier's bank where the amount is converted to the supplier's domestic currency, any currency translation charge deducted and paid to the supplier's bank, and the remainder then credited to the supplier's account. Once confirmation is received through the sensor that the payment has been so credited, the code instructs a Blockchain Ledger to add a new block reflecting the transaction into the ledger.

With the above background and technical discussions providing necessary context, we now shift our focus to address in Section IV below, various legal issues and disputes that are likely to arise

[33] https://docs.accordproject.org/docs/accordproject.html and https://docs.accordproject.org/docs/accordproject-concepts.html.

involving Blockchain Ledgers and smart agreements; and in Section V why arbitration is ideally suited to resolve these disputes. We conclude by identifying and discussing, in Section VI, various considerations for drafting suitable arbitration clauses to use with smart agreements.

IV. Legal Issues and Disputes Likely to Arise

A. Technical Issues That Could Lead to Liability

Bill Gates famously said "[s]oftware is a great combination between artistry and engineering." But like artistry and engineering, perfection is illusive. Smart contracts are nothing more than software code written by humans and are therefore imperfect by their very nature. Any number of issues could arise in the design, development or execution of software code and smart contracts are not immune from such problems. Because technical issues can give rise to legal liability, a few of the more common technical issues associated with smart contracts are outlined below.

1. Design Flaws

Software design is the process by which a programmer translates user requirements into software code. A flawed software design will likely lead to unexpected results and sometimes, catastrophic consequences. Sadly, a design flaw in the software for a new flight-control system on the 737 Max plane was responsible for several recent plane crashes killing 346 people.[34] Another design flaw that caught widespread attention recently occurred when a smartphone app developed for the Iowa Democratic Party was rushed into use with technical and design flaws that caused a significant delay in reporting Iowa caucus results.[35]

While it is unlikely that most design flaws in a smart contract could have such tragic or newsworthy consequences, smart contract design flaws could nonetheless, result in significant financial losses and complex business disputes, among other things.

Flaws could occur anywhere in the design, such as the underlying algorithms or the communications protocol. No matter what the cause, smart contract design flaws can lead to significant issues and therefore liability on any number of theories, such as negligence, product liability, or breach of contract resulting from injury to a participant or third-party proximately caused by a defect in a smart agreement.

To mitigate risks, appropriate steps should be taken both during the development and coding of smart contracts to prevent, detect and remediate design flaws and coding errors. Further mitigation can be achieved by the procurement of adequate insurance coverage against any potential residual exposure.

2. Coding Errors/Bugs

---

[34] David Slotnick, "The DOJ is reportedly probing whether Boeing's chief pilot misled regulators over the 737 Max", Business Insider, February 21, 2010; https://www.businessinsider.com/boeing-737-max-prosecutors-investigation-prosecutors-lied-faa-2020-2.

[35] Ben Popken and Maura Barrett, "Iowa caucus app was rushed and flawed from the beginning, experts say", NBC News, February 5, 2020; https://www.nbcnews.com/tech/security/iowa-caucus-app-was-rushed-flawed-beginning-experts-say-n1131216.

As blockchain technology begins to permeate every industry, the importance of smart contracts will increase dramatically, and the software code supporting those smart contracts will likely control billions of dollars of digital assets.[36] While software development has existed for decades, smart contract development platforms were developed in 2015. Due to the recent development of such platforms, there is a notable absence of developer handbooks relating to smart contracts.[37] In short, the development of smart contracts and associated development platforms are still in their embryonic stages.
While they are likely to mature quickly, no matter what the technology, coding errors can and will happen, and the risk associated with such errors increases as the complexity of the code increases. Like design flaws, coding errors may lead to unexpected consequences and attendant legal liability.

It is currently estimated that the amount of cryptocurrency lost to coding errors is quickly approaching $1 billion. The most well-known of which involves "The DAO", an exploit which we will now discuss.

i. The DAO Incident

Distributed Autonomous Organizations (DAOs) are run by programming code and constitute a collection of Smart Contracts[38] operating independently of any human intervention, as long as ding so covers a DAO's survival costs and provides a useful service to its participant base.[39] A DAO is an early-stage investment fund that lacks a manager. There is an initial funding period during which its participants add funds, typically through what is referred to as a "crowd sale", to a DAO to provide it with resources. Investors vote on which projects to fund with the code implementing the Smart Contracts doing the rest.

On April 30, 2016, a particular DAO called "The DAO" was launched with a 28-day funding window. It raised over $150 Million from more than 11,000 participants. In June 2016, one of its participants exploited a known vulnerability in The DAO's code and drained approximately $53 Million from The DAO into an account which that person controlled. The specific error in the code was known to The DAO's creators, but not remedied in time to prevent the error from being exploited.

The appropriate response to the attack created an interesting dilemma. If "the code is the law", as some smart contract proponents have asserted, what happened was perfectly legal because the code executed as it was intended. As such, some participants in The DAO took the position that

---

[36] Kai Sedgwick, "The Billion-Dollar Quest to Eliminate Smart Contract Bugs", Bitcoin.com, July 12, 2018, https://news.bitcoin.com/the-billion-dollar-quest-to-eliminate-smart-contract-bugs/ .
[37] Yos Riady, "Best Practices for Smart Contract Development", November 10, 2019; https://yos.io/2019/11/10/smart-contract-development-best-practices/ .
[38] Ethereum is a global, open-source, blockchain-based distributed computing platform and operating system (so-called "Ethereum Virtual Machine"), featuring Smart Contract functionality, for building decentralized applications. While blockchains have the ability to process code, most are severely limited in what they can do. Rather than providing a limited set of operations, the Ethereum Virtual Machine allows developers to create whatever applications they want on the Ethereum network, including, e.g., DAOs. See: https://blockgeeks.com/guides/ethereum/.
[39] https://blockgeeks.com/guides/ethereum/.

the transfer did not violate the smart contract itself and instead, exploited a vulnerability in the code. Other participants felt their funds had been stolen and allowing the attack to stand would discourage participants from making future investments.

Ultimately, the Ethereum organization running the code voted to restore the funds to the original investors.[40] Since an error existed in the code, The DAO sought to renegotiate the terms -- though renegotiation being an anathema to Smart Contracts.

3. Inflexibility; Incompleteness

Inherently, smart agreements are inflexible and incomplete. They are neither designed for general use, nor are they suited for it.

If smart agreements are, as some in the field ascribe them to be, "immutable, unstoppable, and irrefutable computer code," that code must declare what will happen as a result of every possible contingency that might occur during the life of the contract.

Smart agreements are inflexible because they rely on executing code that is completely deterministic, i.e., it embodies predefined rules typically reduced to codified "if-then-else" programming statements. Any conduct by the parties that does not fall within the rules is simply ignored. Consequently, the use of smart agreements is usually limited to situations where parties, at the outset of their transactions, can anticipate each and every contingency that might arise affecting their contractual performance. Such transactions tend to be relatively simple as their performance is predicated only on whether a particular condition(s) is satisfied or not, thus being easily translatable into rule(s) of performance which can be readily codified.

But, for many legal contracts that are less simplistic, contractual performance is not so easily assessed because it is not simply whether a predefined logic condition(s) was objectively satisfied or not but rather a determination that requires some degree of human subjectivity. Specifically, the parties or an adjudicator may need to subjectively assess the effect on contractual rights and obligations of the parties resulting from a contingency that occurred and/or prior conduct by one or more of the parties. In those situations, significant portions of the parties' agreement cannot be coded as they are encompassed by non-deterministic concepts and general clauses, such as good faith, reasonableness, intent, excused performance and many other subjective aspects which collectively form the foundation of contract law.[41] Consequently, these legal agreements, by virtue of their nature, are inappropriate for codification and implementation as a smart agreement.

Further, for many such less-simplistic legal contracts, deterministic completeness is unattainable. In practice, it is often extremely difficult, if not impossible, for contract drafters, dealing with anything other than very simple, straight-forward transactions, to anticipate every such contingency that might possibly arise, no matter how small its probability of occurrence.

---

[40] *Ibid*.

[41] Pietro Ortolani, "The impact of blockchain technologies and Smart Contracts: arbitration and court litigation at the crossroads", Uniform Law Review, Volume 24, Issue 2, June 2019 (published by Oxford University Press), p. 438; https://academic.oup.com/ulr/article/24/2/430/5490658 .

Consequently, many commercial legal contracts are incomplete. By leaving certain contingencies and hence their outcomes undefined, the drafters introduce, whether intentionally or not, ambiguities and gaps into commercial legal contracts for later resolution. Oftentimes, it is simply too costly to proceed otherwise. Parties may also recognize and intentionally retain ambiguities and gaps in their legal contracts so that, if a corresponding situation arises later, the parties can then exploit the incompleteness in a way that results in a better ex-ante contract for them. Renegotiation is a common way that ambiguities are resolved and contractual gaps filled.[42] Parties need some degree of flexibility in resolving contractual incompleteness that avoids locking themselves into rigid commitments and outcomes to which they did not anticipate and do not want.[43]

Consequently, for other than relatively simple, completely deterministic transactions, it is quite possible that the code in smart agreements will not reflect some contingencies. Code is not subject to renegotiation. Smart agreements, once they are embodied into code, are fixed with, as some smart agreement adherents vociferously advocate: "The Code is Law", i.e. the code is meant to be the ultimate arbiter of a deal it represents, specifically a stand-alone, self-enforcing agreement not subject to interpretation by outside entities or jurisdictions.[44] If parties decide to modify their smart agreement, they then need to change its code accordingly.

Yet, what happens in a smart agreement if an unanticipated (non-coded) contingency occurs? Does the contract just assume a default or error state, pending some human intervention to clear that state -- which lies directly contrary to the autonomous, self-executing nature of a smart agreement? Should the contract simply report that event to the blockchain and then reset itself once that event ceases and then return to normal execution? At present, there are no definitive answers. When such a situation arises -- as discussed above in the context of The DAO exploit, an errant result can flow from execution of a smart agreement which, in turn, could lead to a dispute between the contracting parties with potentially significant attendant legal liability.

Legal disputes and potential liability can arise, whether under doctrines of negligence, product liability or breach of contractual warranties, where smart agreements are operated beyond their design limits, i.e. under conditions that were not contemplated, particularly where they invoke unintended, possibly even adverse, results.

4. Security Vulnerabilities

Smart agreements are often designed to manipulate and hold funds denominated in Ether, making them tempting targets because a successful attack would result in stealing funds from the contract.[45] While exploited vulnerabilities have captured the headlines and imaginations, recent academic research reported that, out of 21,270 vulnerable smart contracts, at most only 504 have

---

[42] Larry D. Wall, " 'Smart Contracts' in a Complex World", *Notes from the Vault, Federal Reserve Bank of Atlanta,* July 2016; https://www.frbatlanta.org/cenfis/publications/notesfromthevault/1607.aspx.
[43] "What Smart Contracts Need to Learn", *Lawbitrage*, September 4, 2014; https://lawbitrage.typepad.com/blog/2014/09/smart-contracts.html.
[44] "Understanding the DAO Attack", Coindesk, June 25, 2016; https://www.coindesk.com/understanding-dao-hack-journalists .
[45] "Smart Contract Vulnerabilities: Does Anyone Care?", Perez & Livshits, May 17, 2019; https://arxiv.org/pdf/1902.06710.pdf

been subjected to exploits, likely due to the fact that a majority of Ether is held by only a small number of contracts.[46]

While now the number of exploited vulnerabilities may be relatively low, as the technology becomes more widely accepted and more money is exchanged through smart agreements, there can be little doubt that vulnerabilities will be substantially exploited. Such vulnerabilities will therefore expose any number of parties directly or indirectly responsible for the vulnerability to liability including developers, contract administrators, or the entity that hosted the contract.

5. Privacy

Information stored on a Blockchain Ledger may identify aspects of a user's identity and include financial, medical or consumer personal information. Care must therefore be taken to ensure compliance with applicable privacy laws.

Over the last few years, there have been a proliferation of new privacy laws, each one placing more emphasis on the right of consumers to protect their own personal information. The General Data Protection Regulation (GDPR), addressing data protection in the European Union and the European Economic Area, and the California Consumer Privacy Act (CCPA), addressing personal information of California consumers, are recent additions to ever expanding privacy regulations. Both GDPR and CCPA expansively define "personal information" to include any information that directly *or indirectly* identifies a person and therefore could impose significant obligations, as well as risk, on administrators of a Blockchain Ledger to ensure that personal information is properly secured. GDPR and CCPA also present interesting questions about how an individual whose personal information on a Blockchain Ledger can exercise their right to have their personal information deleted (also known as the right to be forgotten under GDPR).

By 2023, Gartner predicts that 65% of the world's population will have its personal information covered under modern privacy regulations, up from 10% in 2020.[47] As such, the privacy and security of personal information on a Blockchain Ledger and/or associated with smart contracts could pose a significant liability.

Consideration should also be given to whether the smart contract is stored on a public, private or hybrid blockchain. Public blockchains are visible to all users, while private blockchains are permission based and visible only to persons or entities with appropriate permissions. Another option is a hybrid blockchain that includes both public and private aspects. Decisions regarding the storage of a smart contract on a public, private or hybrid blockchain may depend on the nature of the information stored.

B. Smart Contracts and Smart Legal Contracts

1. Jurisdiction

---

[46] *Id.*

[47] "Gartner Predicts for the Future of Privacy 2020", January 20, 2020; https://www.gartner.com/smarterwithgartner/gartner-predicts-for-the-future-of-privacy-2020/

Blockchains present a unique jurisdictional challenge that may bar lawsuits that directly involve them. To date, while a small number of lawsuits has been filed that implicate blockchains, these related mainly to claims of securities fraud and misrepresentation in the public sale of an initial coin offerings (ICOs) where the ICOs were to be implemented on blockchains.[48] The authors of this paper are not aware of any lawsuits that yet exist directly concerning transactions that occurred on blockchains themselves or issues surrounding execution of the blockchains themselves; though it is fair to predict that such lawsuits will eventually occur.

For an adjudicator, whether a court or an arbitral tribunal, to consider and rule on a dispute, it is canonical law that the adjudicator must be seized with jurisdiction: over the parties for in personam jurisdiction or an object in dispute for in rem jurisdiction. In either instance, the location of the person or object determines whether jurisdiction arises.

A blockchain is a decentralized structure of information: stored bits of information (code and data) effectively disbursed over many different "locations", as is an entire blockchain infrastructure implemented as "blockchain-as-a-service" (BaaS).

One cannot point to a blockchain or reach out and touch it as it is not physical; it is a data structure: nothing more. It has no physical presence. It is not a physical object. It is an abstraction: a collection of either the presence or absence of electronic charges in separate memory locations respectively representing binary "1s" and "0s" typically accessed by virtualized servers that execute blockchain code and process its data, all residing, often piecemeal, somewhere in a cloud or even across multiple interconnected clouds. Even a virtualized server is nothing more than an abstraction: computer code that, when executed, collectively emulates a physical server. [49] That code too can be stored and executed virtually anywhere on a cloud, or even, like any code, transferred from storage in one location to another so that, rather than executing on one physical host computer, it will execute on another, perhaps half- a-world away. Hence, the traditional notion of a "location", as a physical situs of a person or an object and upon which adjudicators assess jurisdiction, has no meaning for a blockchain.

Consequently, traditional physical measures of national court jurisdiction would fail here. Absent an agreement by the parties conferring jurisdiction on a particular court, no national court could exert requisite physical jurisdiction over a blockchain.

---

[48] *In re Tezos Sec. Litig*., No. 17-CV-06779-RS, 2018 WL 2387845 (N.D. Cal. May 25, 2018) and related litigations: *Baker v. Dynamic Ledger Sols., Inc.*, No. 17-CV-06850-RS, 2018 WL 656012 (N.D. Cal. Feb. 1, 2018); *MacDonald v. Dynamic Ledger Sols., Inc.*, No. 17-CV-07095-RS, 2017 WL 6513439 (N.D. Cal. Dec. 20, 2017); *Okusko v. Dynamic Ledger Solutions, Inc. et al.*, Case No. 17-cv-6829; *GGCC, LLC v. Dynamic Ledger Sols., Inc.*, No. 17-CV-06779-RS, 2018 WL 1388488 (N.D. Cal. Mar. 16, 2018); *see also, e.g. Rensel v. Centra Tech Inc.*, et al., 17-cv-24500-JLK (S.D. Fla.); *Hodges, et al. v. Monkey Capital, LLC, et al*., 17-81370 (S.D. Fla.); *Balestra v. ATBCOIN, LLC*, et al., 17-10001 (S.D.N.Y.); *Stormsmedia, LLC v. Giva Watt, Inc., et al*., 17-00438 (E.D.Wash.); *Davy, et al. v. Paragon Coin, Inc., et al*., 18-00671 (N.D.Cal.). Also, for SEC concerns regarding on ICOs, see https://www.sec.gov/ICO.

[49] As the concept of hardware virtualization is well beyond the scope of this paper, it will not be addressed in any detail. For further insight, the reader is referred to any virtualization software provider, such as, e.g., VMWare Inc. (https://www.vmware.com/) and Microsoft Corporation (https://docs.microsoft.com/en-us/windows-server/virtualization/virtualization.)

2. Legal Enforceability: ESIGN; UETA and other state statutes

Both the "Electronic Signature in Global and National Commerce Act" (ESIGN)[50] and the "Uniform Electronic Transactions Act" (UETA)[51] were enacted to help ensure the validity of electronic contracts and the defensibility of electronic signatures. UETA, currently enacted in 47 states, Puerto Rico, the US Virgin Islands and the District of Columbia, provides the states with a framework for determining legality of an electronic signature in both commercial and government transactions. Washington State, New York and Illinois have not yet enacted UETA; however, similar legislation governing electronic transactions has been enacted in each of these three states. UETA is limited to electronic contracts related to business, commercial (including consumer) and governmental matters.

Effective since October 1, 2000, ESIGN accords, as does UETA, electronic signatures and records the same legal status as manually inked signatures and paper-based records. ESIGN only affects the medium through which a contract is made and does not change the underlying substance of any law within its scope. It treats commercial and consumer transactions differently: for commercial transactions, intent to enter into an electronic contract is implied from the surrounding facts and circumstances or by an express statement of intent; while for consumer transactions, it requires the consumer to receive specific disclosures before agreeing to proceed electronically. ESIGN, being federal, affects inter-state commerce.[52] Though ESIGN will pre-empt any inconsistent state law, it expressly precludes preemption of UETA in any state or territory that enacted the latter.[53]

UETA, in contrast to ESIGN, has no consumer notice provision, though certain enacting states have enacted their own variations to UETA to include, among other aspects, such notice. Further, unlike ESIGN, UETA addresses when an electronic record has been sent and received.[54]

The provisions of both UETA and ESIGN are very liberal to encourage adoption and use of electronic contracting. Nevertheless, to the extent contract formation occurs through a Smart Legal Contract rather than through a separate preliminary interaction between the parties, it may be necessary to ensure the contract fully complies with these acts.

By contrast, Smart Contracts, which, as discussed, involve nothing more than providing incoming data (including measured values) to coded logic to correspondingly condition the execution of a blockchain entry, do not implicate electronic formation of contractual obligations. Those obligations are previously agreed to by the parties involved before being defined in code. Accordingly, Smart Contracts are not likely to implicate these and similar acts.

In addition, during 2019, some states enacted legislation specifically enabling the use of Blockchain Ledgers for use in smart agreements or for storing certain records (Illinois -- May 29,

---

[50] 15 U.S.C. §7001, et seq (2000).
[51] Approved and recommended by the Uniform Law Conference in 1999 for state enactment.
[52] https://rightsignature.com/legality/ueta-act
[53] https://www.dlapiper.com/en/us/insights/publications/2018/05/esignature-and-epay-news-and-trends-1-may-2018/a-short-primer-on-applicable-us-esignature-laws/ .
[54] https://www.dlapiper.com/en/us/insights/publications/2018/05/esignature-and-epay-news-and-trends-1-may-2018/a-short-primer-on-applicable-us-esignature-laws/.

2019, Maryland -- April 30, 2019, Nevada -- June 7, 2019 and Texas -- June 10, 2019) or have established a task force to implement and expand the blockchain industry in that state (Florida -- May 23, 2019). Other states have amended their state UETA Acts to recognize blockchain technology (North Dakota and Oklahoma -- both late April 2019. and Nevada -- June 7, 2019).[55]

V. Arbitration -- The Only Viable Approach for Blockchain Disputes

If traditional courts and arbitral tribunals lack jurisdiction to hear these disputes, then who or what will?

Once blockchain technology achieves sufficient widespread commercial use, disputes involving blockchain technology will inevitably arise. What is needed is a fast, inexpensive, transparent and reliable arbitral system, having decentralized jurisdiction across an entire blockchain, that renders ultimate judgments.

Currently, there are no uniform standard arbitration procedures for arbitrating disputes involving smart agreements.[56] These technologies are simply too new. Developmental efforts are underway in the field to provide fully automated arbitral platforms for use with blockchains. One example, which relies on game theory, is the "Kleros" platform which executes on the Ethereum network as an autonomous organization[57]. Another approach, which recognizes the necessity of human decision-makers, is embodied in the "Codelegit" arbitration library. That library provides a set of coded provisions that can be incorporated into a Smart Legal Contract to principally integrate a traditional arbitral proceeding into the contract and allow either party to pause, resume, modify and end the contract. A resulting award is then applied as input to the Smart Contract to establish a new transaction on the blockchain to self-enforce the award.[58]

Such platforms may ultimately prove useful in time-and cost-effectively resolving simple, straightforward disputes where rule-based economic analyses suffice. Many legal disputes however require, to reach a "just" result, subjective analysis by skilled, knowledgeable human decision-makers familiar with the industry and commerce at issue, the technology and the underlying law, who render decisions not dictated reflexively by rules or algorithmic predictions but on their own wisdom built up through years of experience. There, such automated platforms may prove to be inadequate.

---

[55] Margo H.K. Tank et al, "Blockchain and Digital Assets News and Trends," *DLA Piper Publications*, May 24, 2019 and June 24, 2019; respectively: https://www.dlapiper.com/en/us/insights/publications/2019/05/blockchain-and-digital-assets-news-and-trends-may/; and https://www.dlapiper.com/en/us/insights/publications/2019/06/blockchain-and-digital-assets-news-and-trends-june/.

[56] Sara Hourani, "The Legal Reality of the Recognition and Enforcement of Cross-Border Blockchain-based Arbitral Awards: Beyond Futuristic Idealism?" *Off the Chain* (newsletter), May 18, 2019; https://www.odrblockchain.com/off-the-chain/2019/001/the-legal-reality-of-the-recognition-and-enforcement-of-cross-border-blockchain-based-arbitral-awards-beyond-futuristic-idealist.

[57] Clement Lesage et al, *Kleros, Short Paper v 1.0.7,* September 2019; https://kleros.io/assets/whitepaper.pdf

[58] Morgane Guyonnet, "CodeLegit White Paper on Blockchain Arbitration"; https://docs.google.com/document/d/1v_AdWbMuc2Ei70ghITC1mYX4_5VQsF_28O4PsLckNM4/edit. See also http://codelegit.com/blog/.

As such, an effective practical approach may well be for blockchain administrators to impose a contractual framework onto all their participants to which each participant would assent as a condition for joining the blockchain. That framework would: specify a certain arbitral forum (e.g., AAA/ICDR or other institution) to which participants would bring their disputes for resolution and which would have sufficient power to enforce all resulting resolutions, define a specific process, set forth a governing rule set, and define or reference governing substantive law.[59]

Aside from arbitration overcoming the principal obstacle to national litigation: jurisdictional limits caused by the decentralized nature of blockchains, arbitration presents the following other distinct advantages over litigation which uniquely render arbitration ideal for resolving blockchain-based disputes.

[59] An interesting parallel to this framework is the Uniform Domain Name Dispute Resolution Policy (UDRP) and its associated Rules, both adopted by ICANN (Internet Corporation for Assigned Names and Numbers) on October 24, 1999, used to redress cybersquatting of domain names (for certain generic top level domains, such as , e.g., .com, .edu..org and various country codes). The UDRP is a voluntary alternative to national court adjudication, The UDRP specifies, for example, in paragraph 4: substantive provisions that collectively constitute prima facie cybersquatting; enumerates, with reference to the Rules, a summary arbitral procedure; and defines limited relief (cancellation or transfer) available to prevailing complainants. Domain name registrants, whenever situate in the world, contractually agree to be bound by the UDRP as a condition of registering their domain names at accredited registrars. Those registrars also agree, through their accreditation agreements with ICANN, to implement the UDRP as a necessary condition of accepting registrations.
Further, this framework could be implemented by a global industry-wide consortia which might also, illustratively:
(a) define interoperability standards of software components of BaaS and other blockchain infrastructures and also of APIs (application programming interfaces) between legacy software systems and blockchain infrastructure to facilitate and expedite development and commercial exploitation of blockchain technology, and permit competitive offerings of infrastructure software components;
(b) certify, based on those standards, operability and robustness of internal components for BaaS infrastructures to promote their adoption and use, and
(c) define and promulgate a scheme for accrediting arbitral institutions to provide dispute resolution services under the framework.

A. Protection of Proprietary Information

Protection of proprietary information is not only important to the parties, but it is also important to arbitral institutions and its neutrals.

Confidentiality is an important feature of arbitration. The American Arbitration Association (AAA), for example, imposes upon its staff and neutrals an ethical obligation to keep information confidential.[60] In any arbitration, regardless of the arbitral institution, the parties maintain their right to disclose details of the proceeding, unless they have a separate confidentiality agreement in place.

Maintaining the privacy and security of personal information is also a very important aspect of arbitration. Arbitral institutions now have policies to address their role in securing personal information. AAA and its international division, the International Center for Dispute Resolution (ICDR) has, for example, implemented best practice policies, technologies and procedures to help protect its data and technology resources.[61] The policy requires AAA-ICDR employees to attend annual security awareness training, and compliance audits are conducted. Regular audits and system tests are performed to ensure compliance with security related polices. Arbitrators are also now addressing information security during the preliminary hearing with parties and/or their representatives.

B. Specialized Knowledge of the Tribunal

Not only is arbitration more efficient and cost-effective than litigation, but it gives parties involved in the dispute the opportunity to select their arbitrator(s), giving all parties confidence that an equitable solution will be reached.[62]

Given the complexity of the underlying technology and likelihood of technical issues, it is therefore important to ensure that the tribunal addressing these disputes has specialized knowledge or expertise. Because software development is an integral part of the smart contract, an arbitration clause relating to a smart contract dispute should include a clause requiring arbitrators to have experience in software development.

C. AAA Procedural Flexibilities

1. Formulation of Specific AAA/ICDR Rule Set for Smart Contract and Smart Legal Contract Disputes

An arbitral process is remarkably open-ended and relatively informal: a blank canvas on which parties can collectively create the exact process they need and no more. Under the AAA

---

[60] *AAA Statement of Ethical Principles*", https://www.adr.org/StatementofEthicalPrinciples
[61] "ICDR Secure Case Administration", https://www.icdr.org/Secure_Case_Administration; *see also* "AAA-ICDR® Information Security Program" https://adr.org/sites/default/files/document_repository/AAA_InformationSecurity_Summary.pdf
[62] "Arbitrators Provide Technical Expertise, Confidentiality", Jean Baker, Corporate Counsel Business Journal, January-February 2020 .

Commercial Arbitration Rules, parties are completely free and have total autonomy to decide what specific steps they will use and when, and all related aspects, subject only to affording mutual due process. These rule sets, while sufficiently definite and inclusive to define a minimal but essential framework of an arbitral process that can yield a legally binding award, are intentionally very broad and quite malleable to provide parties with sufficient latitude to exquisitely adapt the process to fit the characteristics of their dispute. In effect, the parties can thoughtfully and deliberately "fit the process to the fuss", thus crafting the arbitral process to nicely conform to the characteristics of their blockchain-related smart agreement disputes.[63]

In some instances, successive blockchain transactions can occur rather quickly. Consequently, to be effective and prompt, an arbitral proceeding must be focused and rather short: reduced, as much as possible, to its essential elements to render an award in a manner that minimizes adverse impact on future incoming transactions.

Dramatically limiting the available time during which the proceeding occurs forces counsel to sharply concentrate their efforts from the onset on the core issue(s) in contention, excluding all secondary and tangential issues from discovery, briefing, motions, and the hearing itself. Where very little time is allotted for arbitration, all discovery and motion practice may well be eliminated altogether. As discovery costs are often the largest cost-driver in an arbitration, its elimination alone can yield significant cost savings. Further, a short process time may only permit the merits hearing to consume no more than a few hours: a morning or an afternoon.

An emergency arbitration is such a proceeding. The proceeding is defined in Article 6 of the International Arbitration Rules of the ICDR (International arm of the American Arbitration Association - AAA) and Rule 38 of the AAA Commercial Arbitration Rules. An emergency proceeding can yield an award in no more than a few weeks, and, with the proceeding further condensed in time, in just a few days.

As the needs of some blockchain disputes involving smart agreements may, to a considerable extent, parallel those of disputants seeking emergency relief, the AAA/ICDR emergency arbitration rules provide a particularly germane starting point for developing a rule set designed to handle disputes involving smart agreements.

D. Procedural Considerations

In a blockchain-related smart agreement dispute, much, if not all of the evidence, and most, if not all, the arbitration submissions from the parties will reside as separate transactions stored on the blockchain itself. Consequently, arbitrators hearing such disputes must be provided with secure, read access to all salient (if not every) stored transactions on the blockchain. This requires that the arbitrators be provided with appropriate client software to securely access, read and copy transaction information from individual blocks along with whatever permissions, cryptographic keys and/or other credentials are necessary to properly use that software.

---

[63] Peter L. Michaelson, "Patent Arbitration: It Still Makes Good Sense", *Landslide* (publication of the ABA Section of Intellectual Property Law), Vol. 7, No. 6, July/August 2015, p. 46.

Further, to provide arbitrators with the ability to see, not just hear, witnesses and hence make more accurate assessments of credibility, arbitrators and parties may choose to eliminate traditional in-person or even telephonic hearing modalities in favor web-based multi-site videoconferencing. Reliance on purely electronic modalities also advantageously eliminates travel cost and time, thus furthering the goal of providing an effective, efficient and rapid proceeding.

VI. Arbitration Clauses

As smart contracts are written in software code, they lack the typical clauses found in most legal contracts which establish the foundation for an arbitration, such as the consent to arbitrate, seat of arbitration, governing law, arbitral institution and governing rules. That does not however mean that such clauses do not apply to the arbitration of smart contracts. In fact, they do.

As previously discussed in Section II(B) of this paper, a Ricardian Contract or a smart legal contract, that includes both a "smart" (computer-executed) and "non-smart" (traditional text-based) clauses, allows parties to address all necessary contract terms well in advance of a dispute.

A. Consent to Arbitrate

Article II of the 1958 New York Convention on the Enforcement of Foreign Arbitral Awards (the "Convention") requires that agreements to arbitrate be in writing. It defines the term "agreement in writing" to be an arbitral clause in a contract or an arbitration agreement, signed by the parties or contained in an exchange of letters or telegrams.

Smart contracts are, however, nothing more than software code which usually only a programmer fully understands. It would therefore be nearly impossible to meet the consent to arbitrate requirements of the Convention without a text-based contract that is used as a companion to a smart contract.

B. Arbitral Seat

The framework for the arbitration is established by the arbitral seat. Selection of the seat will have practical and legal consequences. For example, the law of the seat provides the procedural law for the arbitration, including, *inter alia*, tribunal's authority, powers, and duties. It also establishes the court where an award may be challenged.

Because smart agreements are geographically distributed by nature, it is important to consider the practical and legal effect a seat may have on the dispute being arbitrated. Given the novelty of smart agreements, parties should fully consider how the arbitral seat may affect the dispute and specifically consider whether smart agreements are legal, enforceable and arbitrable in the seat and that awards can be enforced. Once consideration is given to those factors, the seat can be specified accordingly.

C. Enforceability

Unless and until there is sufficient participant confidence and legal clarity in the enforceability of a Smart Legal Contract -- whether in the United States or elsewhere, parties intending for their underlying transactions to have legally binding effect should consider incorporating arbitral clauses, governance and/or automatic enforcement mechanisms to limit circumstances in which they will require judicial intervention or to facilitate enforcement of arbitral or judicial decisions.

For example, parties or the blockchain platform itself may include an escrow procedure. The parties also may build into their Smart Legal Contract mechanisms to stop automatic performance of the contract should a dispute arise or, alternatively, mechanisms to permit the return of funds or other assets by providing access to Smart Legal Contracts to certain accounts funded by the parties.

Contracting parties also may consider utilizing blockchain platforms that contain alternative dispute resolution mechanisms, such as suspension of the contract pending resolution coupled with automatic referral of a dispute to the AAA/ICDR for resolution. Even with any such contractual mechanism, it is however likely that a need will still remain for some degree of judicial review and/or enforcement of any ensuing arbitral award or compulsion of a third-party to participate in an arbitral proceeding, thus precluding a totally automatic, self-executing arbitral process forsaking any judicial involvement whatsoever.[64]

Further, arbitral awards rendered in any signatory member state are enforceable, under the provisions of the Convention and subject to its conditions, in approximately 160 other signatory member states.

As the concept of awards for Smart Legal Contracts, produced through automated blockchain technology, is quite novel, a question invariably arises as to whether these awards constitute a valid award for purposes of enforcement under the Convention and particularly by national courts of its member states.

Article I of the Convention is silent on any specific form an arbitral award must take, including whether it be in written form or not or in a specific format to be signed by the arbitrators. Article VII(1) encourages other multilateral or bilateral state agreements on the recognition and enforcement of arbitral awards to take precedence over the provisions of the Convention in order to encourage recognition and enforcement of foreign arbitral awards. Hence, it is likely that, under the Convention, a blockchain-based award, authenticated in code, may be considered valid, though the authors are not presently aware of any ruling from a court or other forum which addressed the issue.[65]

Assuming the Convention per se presents no evident limitation to recognizing and enforcing such awards, then the focus shifts from the Convention to national legislation which might.

In that regard, the Convention contains provisions that often refer judges back to the application of relevant domestic law. For example, a national court may refuse to recognize and/or enforce an arbitral award if, under Article V(1)(e), it has not yet become binding on the parties or has been set aside or suspended by the competent court at the seat of arbitration, or under Article V(2)(b), it lies contrary to public policy of that nation. Consequently, Article V may limit recognition and enforcement of blockchain-based Smart Legal Contract awards, that are

---

[64] "Smart Contrasts: Is the Law Ready?". Smart Contracts Alliance, Chamber of Digital Commerce, September 27, 2018, p. 31; https://digitalchamber.org/smart-contracts-whitepaper/.

[65] Sara Hourani, "The Legal Reality of the Recognition and Enforcement of Cross-Border Blockchain-based Arbitral Awards: Beyond Futuristic Idealism?" *Off the Chain* (newsletter), May 18, 2019; https://www.odrblockchain.com/off-the-chain/2019/001/the-legal-reality-of-the-recognition-and-enforcement-of-cross-border-blockchain-based-arbitral-awards-beyond-futuristic-idealist.

only authenticated in code, if those awards are invalid under applicable national law at their seats of arbitration or their places of enforcement.

So far, the current legal framework under the Convention appears to allow for recognizing and enforcing blockchain-based arbitral awards if they are valid under the law at the seat of arbitration and/or the place of enforcement.

Clearly, over time, some jurisdictions will likely be more willing to recognize and enforce these novel forms of arbitral awards than others are. It remains to be seen, once appropriate jurisprudence starts appearing from the former jurisdictions, just how open and to what extent they will be and what conditions, if any, they will impose.

D. Governing Substantive Law

The parties to an arbitration are free to contractually select, in their arbitration agreement, whatever body of substantive law they want to govern their arbitration. This is done by specifying, through a choice of law clause, the substantive law of a jurisdiction, preferably a jurisdiction having a long-term, consistent, fair and well-developed body of commercial jurisprudence on which the parties can reasonably rely throughout their contractual relationship. For that reason, the substantive law of well-known jurisdictions, such as the States of New York and California, are often used, as is English law. A choice of law clause should exist in any agreement underlying a Smart Contract and also directly within a Smart Legal Contract itself.

E. Incorporation of Arbitral Institution and Governing Rule Set

Similarly, through an appropriate clause in their arbitration agreement, the parties are free to contractually select whatever institution they desire to administer their arbitration and whatever rule set they choose out of those then provided by the institution. They should have such a clause in any agreement underlying a smart agreement and also directly within a Smart Legal Contract itself.

Illustratively, in many contracts, the parties specify the AAA and select its Commercial Arbitration Rules then in effect. Alternatively, parties can also choose to arbitrate on an "ad hoc" basis, i.e. having the arbitral tribunal rather than an institution completely administer the proceeding, and often do so to save institutional filing fees and other costs. The present authors, based on their extensive arbitral experience, view ad hoc arbitration as short-sighted as the advantages obtainable through institutional administration[66], often significantly outweigh whatever cost savings an ad hoc process might provide, let alone when in a complex and time-sensitive proceeding as a block-chain related arbitration is likely to be.

VII. Conclusion

[66] Benefits of institutional administration include, for example, an existing panel of skilled arbitrators with arbitral, legal and technical expertise and experience; effective and efficient case management; financial oversight and management; separation and insulation of the arbitral tribunal from discussions with the parties concerning arbitral fees and financial status of each party; and reliance on the institution for appropriate guidance by the Tribunal and the parties.

Blockchain Ledgers, in light of the immutable trust and security they provide, and, by extension, smart agreements which incorporate these ledgers, are an evolutionary technology that is destined, over the coming years, to experience rapidly and widely expanding use throughout many diverse fields. Through that use, disputes will inevitably arise. Arbitration offers a highly practical, if not the only realistic, way to efficiently and effectively resolve them.

Smart Contracts & International Arbitration

Article *in* SSRN Electronic Journal · January 2018

DOI: 10.2139/ssrn.3290026

CITATION
1

READS
1,897

1 author:

Ibrahim Shehata
Cairo University
5 PUBLICATIONS **1** CITATION

SEE PROFILE

SMART CONTRACTS &

INTERNATIONAL ARBITRATION

Ibrahim Mohamed Nour Shehata

PhD Candidate at Maastricht University, Lecturer Assistant at Cairo University

Section (A): Introduction to Smart Contracts

A blockchain is simply a decentralized ledger for recording digital data in a verified time-stamped manner without the need for a trusted third party.[1] Blockchain technology provides more "security, traceability, and transparency of records…as well as lower operational costs." [2] In this regard, public blockchains are protected from security threats because they maintain the information on multiple nodes where more than 51% of the nodes would have to be compromised before any security breach could occur.

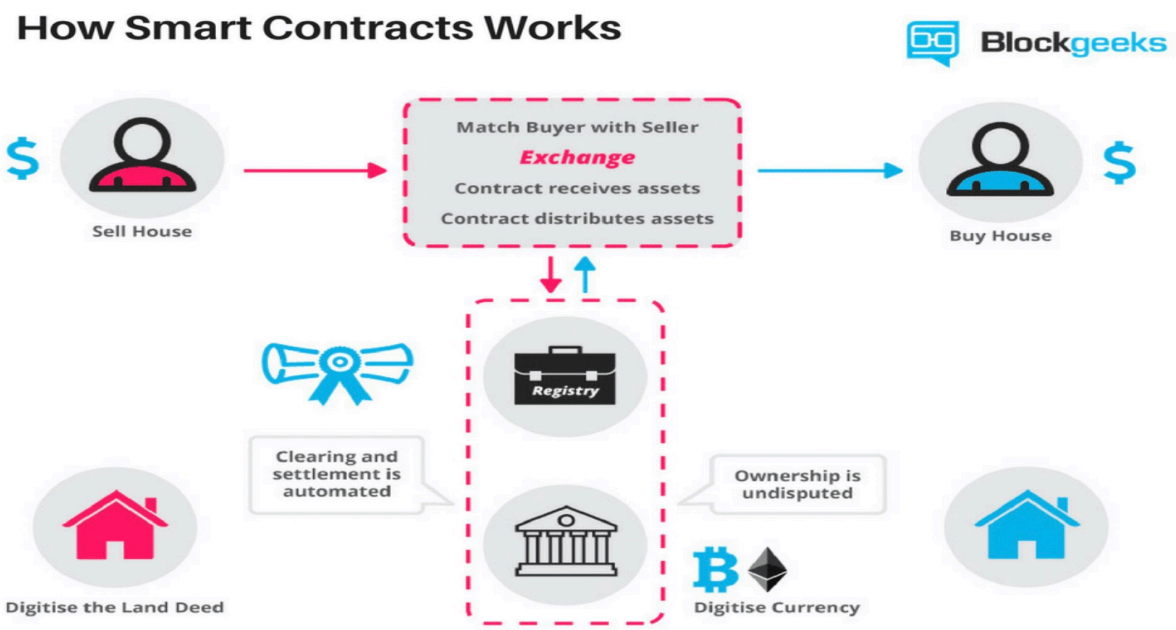
The term "smart contracts" usually refers to software programs that are built on the blockchain to execute agreements reached by the parties. In the late 1990s, the computer scientist Nick Szabo envisioned the use of more robust cryptographic protocols to be used to write computer software that resembled "contractual clauses" in a way that would make the breach of such contracts extremely expensive.[3] Then in 2004, Ian Grigg came up with the notion of a "Ricardian Contract" which is a contract that is readable by both machines and humans.[4] More recently, in 2012, Harry Surden, proposed the concept of data-oriented contracts and the creation of "computable contracts."[5]

1 M. Raskin, *Realm of The Coin: Bitcoin and Civil Procedure.* Fordham Journal of Corporate & Financial Law, vol. 20, no. 4, pp. 969.
2 J. Bambara, et. al, *Blockchain: A Practical Guide to Developing Business, Law, and Technology Solutions* (2017), pp. 101.
3 Nick Szabo, *Formalizing and Securing Relationships on Public Networks.* Available at http://ojphi.org/ojs/index.php/fm/article/view/548/469.
4 Ian Grigg, *The Ricardian Contract*, available at http://iang.org/papers/ricardian_contract.html.
5 Harry Surden, *Computable Contracts,* University of California-Davis Law Review 46 (2012): pp. 629.

The introduction of blockchains such as Ethereum made it possible to implement the ideas first envisioned by Nick Szabo over twenty years ago. In this regard, parties can use Ethereum to enter into a smart contract which resembles a binding commercial relationship. Such a contract can be entirely or partially recorded using code. Also, such code can be used to manage the contractual performance of the parties to the smart contract.

The best definition of a smart contract is: "a set of promises, specified in digital form, including protocols within which the parties perform on these promises."[6] Accordingly, a smart contract is a computerized algorithm which automatically performs the terms of the contract. Smart contracts lie on a wide spectrum ranging from vending machine contracts to fully blockchain-executed smart contracts.[7] A recent example of fully blockchain-executed smart contracts is a smart contract for a flood insurance policy, linked to the precipitation data from the Met Office. Once the data from the Met Office feeds into the blockchain, the policy is automatically triggered, and insurance claims are paid out.[8] Our discussion in this paper will focus on smart contracts executed on public blockchains such as Ethereum. Please find below a chart explaining the concept of smart contracts that are executed on blockchains.[9]

6 Nick Szabo, *Smart Contracts: Building Blocks for Digital Markets* (1996).
7 Bambara, *Blockchain: A Practical Guide to Developing Business, Law, and Technology Solutions* (2017), pp. 101.
8 Available at: http://www.nortonrosefulbright.com/knowledge/publications/157162/arbitrating-smart-contract-disputes
9 Available at: https://blockgeeks.com/guides/smart-contracts/



Section (B): Smart Contracts v. Traditional Contracts

Smart contracts typically have the following characteristics: (1) execution of the smart contract is automated;[10] and (2) performance of the smart contract is ensured without recourse to the courts.[11] In this regard, the main difference between smart contracts and traditional legal contracts is "the ability of smart contracts to enforce obligations by using autonomous code."[12] Smart contracts do that by recording performance obligations in a strict and formal programming language (like Ethereum's Solidity).

Generally speaking, the code of the smart contract is executed without relying upon a trusted third party[13]; the code is rather implemented in a distributed manner by all of the nodes supporting the

10 M. Raskin, *The Law and Legality of Smart Contracts* (2017). pp. 306.
11 Ibid.
12 De Filippi Primavera and Aron Wright, *Blockchain and the Law: The Rule of Code* (2018).
13 Please see below our recommendation for the inclusion of oracles in smart contracts, whereby we argue that this hypothesis is overestimated when it comes to smart contracts dealing with off-the-chain events.

underlying blockchain-based network whereby no single party controls the blockchain[14] (i.e., Ethereum). This autonomous scheme makes the promises recorded into smart contracts to be - by default - more difficult to get amended or terminated than promises in traditional legal contracts recorded in natural language (i.e., legalese). Accordingly, unless the parties have incorporated some logic in their smart contract to enable the amendment and the termination of such a smart contract, then there might be no way to halt the execution of a smart contract after it has been triggered by its parties.[15]

Section (C): Smart Contracts Legal Challenges

Smart contracts raise numerous enthralling legal challenges. This section will try to shed light upon some of these legal challenges as follows:

i. Legal Effects:

As a starting point, are smart contracts legal binding contracts? The answer to this question depends upon three main factors: (1) the specific use case; (2) the form of smart contract being used (i.e. entirely coded in software or a hybrid smart contract with both an encrypted coded version and a text-based version); (3) the law applicable to the contract. This means that the answer might vary significantly depending on the concerned jurisdiction. Often, the certainty of the content of the contractual terms and whether they are comprehensive enough is a critical factor in determining the legal effects of any contract in numerous jurisdictions.[16]

14 This is the case with public blockchains only. There are private blockchains which are usually administrated and controlled by a trusted third party.
15 Kevin D. Werbach and Nicolas Cornell, *Contracts Ex Machina*, Duke Law Journal 67 (2017).
16 Bambara, *Blockchain: A Practical Guide to Developing Business, Law, and Technology Solutions* (2017), pp. 103-4.

In order to eliminate such uncertainty surrounding the legal effects of smart contracts, some states like Delaware, Tennessee, and Arizona have passed legislation to recognize the legal effects of smart contracts. In 2017, Arizona passed the amended Arizona Electronic Transactions Act (AETA), HB 2417, which defines blockchain technology as a "distributed, decentralized, shared and replicated ledger, which may be public or private, permissioned or permission less, or driven by tokenized crypto economics or token less" [17] and indicates that the "data on the ledger is protected with cryptography, is immutable and auditable and provides an uncensored truth." [18] HB 2417 also defines smart contracts as an "event-driven program, with state, that runs on a distributed, decentralized, shared and replicated ledger that can take custody over and instruct transfer of assets."[19]

This means that Arizona recognizes the legal binding effects of smart contracts that are fully automated and executed on a blockchain, even if there is no corresponding traditional contract in word-format. Therefore, parties to a smart contract might be able to ensure that their smart contract is legally binding if they elect the law applicable to the contract to be that of Arizona, or Delaware or Tennessee or any other jurisdiction that recognizes the legal binding effects of smart contracts. Such a choice of law has to be supplemented by choice of forum that would recognize and enforce the parties' choice of law. As will be discussed below, the favorable forum in this respect would be international arbitration that has its seat in Arizona, or Delaware or Tennessee or any other jurisdiction that recognizes the legal binding effects of smart contracts.

17 Ibid.
18 Ibid.
19 Ibid.

ii. Amendment and Termination of Smart Contracts:

The original smart contract concept has started with the invention of the vending machine. With a vending machine for soft drinks, one can insert a dollar for instance and gets back a soft drink. However, the process of a vending machine is not flawless. For instance, what if one changed his mind after inserting the dollar and wants to get chocolate instead of a soft drink; or, what if one changed his mind and did not want anything anymore. An even more intriguing question, what if the vending machine does not perform its obligation and dispenses the soft drink; I am sure many of us have faced such a situation and did not know what to do. These examples also apply in the realm of smart contracts which are entirely recorded on blockchains. As will be discussed below, the Ethereum platform experienced a technical "hard fork" response in order to be able to unwind the effects of smart contracts on the decentralized autonomous organization ("**DAO**").[20]

iii. Coding limitations:

Whenever one mentions coding limitations in the world of the blockchain, the DAO incident has to be mentioned. The DAO was formed in 2016 to create an investing fund that "would not be controlled by any one individual, but by shareholders voting based on their stakes on a blockchain."[21] The DAO was able to pool funds worth $150 million. Soon after this money was raised, a hacker was able to divert about what is worth $40 million funds from the DAO in an unpredictable manner. The hacker did not "hack" the code in a malicious way but rather exposed a legal loophole in the smart contracts of the DAO.[22] This incident shows how coding is limited and how bugs could be simply exploited by hackers. Thus, it is not really surprising that a 2016 study of Ethereum smart contracts revealed that

20 Ibid.
21 Raskin, *The Law and Legality of Smart Contracts* (2017). pp. 337.
22 Ibid.

there are at least 100 errors per 1,000 lines of code.[23] An intriguing question that would arise in this context would be, who should be liable for such mistakes or errors? [24] In traditional contracts, the parties would be able to sue the drafting lawyer for malpractice, could a similar lawsuit be brought against the coders of smart contracts for coding errors. These are novel legal issues that do not exist with traditional text-based contracts; it will be interesting to see how courts and arbitral tribunals will deal with such incidents.

iv. Ability to design complex contracts:

As the adoption of blockchain spreads, smart contracts will become increasingly complex and capable of handling highly sophisticated transactions. Currently, coders are already stringing together multiple transaction steps to form more complex smart contracts. Nonetheless, we are many years away from code being able to determine more subjective legal criteria. For instance, there is no yet code that would be able to determine whether a party satisfied a commercially reasonable efforts standard or whether a force majeure clause should be triggered or not. [25]

Section (D): Recommendations for the Future Landscape of Smart Contracts

i. Un-Anonymizing the Identity of the Parties to Smart Contracts:

From a purely legal perspective, having a contract entered into by pseudonymous parties raises more than one question. First and foremost, how would one be able to validate the capacity of such parties to enter into smart contracts in the first place? Also, what if both parties wanted to amend their agreement to be in line with the new economic conditions or amend it for any reason; would the

23 David Zaslowsky, *What to Expect When Litigating Smart Contract Disputes*, Law360 April 4, 2018.
24 Bambara, *Blockchain: A Practical Guide to Developing Business, Law, and Technology Solutions* (2017), pp. 103-4.
25 Stuart D. Levi and Alex B. Lipton, *An Introduction to Smart Contracts and Their Potential and Inherent Limitations*, Skadden, Arps, Slate, Meagher & Flom LLP and Affiliates, May 7, 2018. Available at https://www.skadden.com/insights/publications/2018/05/an-introduction-to-smart-contracts

parties be able to do so if they do not even know the identity of each other. What if one of the parties thinks there is a force majeure that should allow him to terminate the smart contract? Would such a party be able to proceed with such an argument if he does not even know the identity of his counter party[26] This party cannot even file a lawsuit; against whom will he file such a lawsuit. Even if such a party were able to obtain a default judgment (against "John Doe" for instance), such a default judgment would not be of much use or effect as long as the identity of John Doe remains unknown.[27]

Therefore, if any platform for smart contracts desires to gain business traction, then it should start by un-anonymizing the parties to smart contracts to enable a desirable degree of flexibility that is required in any commercial contract, whether smart or text-based. This recommendation goes hand in hand with the recommendation for providing the parties with the tools to amend, terminate and unwind their smart contracts. They both work together towards achieving a feasible structure for smart contracts that could be a reliable substitute for traditional contracts.

ii. Enabling the Amendment and Termination of Smart Contracts:

Public blockchains, as we understand, are immutable; this makes amending or terminating a smart contract on a public blockchain a far more complicated process than modifying any software code. This could result in (1) yielding higher transaction costs; and (2) increasing the margin of error for effectuating such amendments. Further, smart contracts do not yet offer analogous self-help remedies similar to those available under traditional contracts. For instance, under a traditional contract, a party can engage in the so-called "efficient breach," i.e., knowingly breaching a contract and paying the

26 Primavera and Wright. *Blockchain and the Law: The Rule of Code* (2018).
27 Ibid

resulting damages if it determines that the cost to perform is greater than the damages it would owe.[28] This is simply not available under smart contracts. That's why there are currently projects underway to create smart contracts that are amendable and terminable at any time. This is certainly "antithetical to the immutable and automated nature of smart contracts; it reflects the fact that smart contracts only will gain commercial acceptance if they reflect the business reality of how contracting parties act."[29]

iii. The inclusion of Oracles in Complex Smart Contracts:

The promise of smart contracts as a decentralized mechanism for contracting is extremely overestimated and overhyped. This promise is true only when all the obligations resulting from the smart contract will take place on the blockchain ("on-the-chain"). If inputs are rather required from the real world ("off-the-chain"), then the promise of decentralization will completely evaporate in the air. In addition, supporting the process of completely "on-the-chain" smart contracts especially concerning dispute resolution would also require a trusted third party. Fortunately, there is a solution; use a trusted third party or what is commonly referred to as an "oracle." [30]

Oracles can be individuals or programs that store and transmit information from "off-the-chain," thereby providing a means for blockchain platforms to interact with real-world persons and potentially react to such external events. For example, oracles can be connected to a data feed from a third party conveying the latest London Interbank Offered Rate (LIBOR). Also, we can make an oracle convey

28 Stuart D. Levi and Alex B. Lipton, *An Introduction to Smart Contracts and Their Potential and Inherent Limitations*, Skadden, Arps, Slate, Meagher & Flom LLP and Affiliates, May 7, 2018. Available at https://www.skadden.com/insights/publications/2018/05/an-introduction-to-smart-contracts

29 Ibid.

30 Alec Liu, *Smart Oracles: Building Business Logic with Smart Contracts*, Ripple, July 16, 2014. Available at https://ripple.com/insights/smart-oracles-building-business-logic-with-smart-contracts/.

the insights of human beings or support private dispute resolution and private arbitration systems.[31] With oracles, smart contracts can respond to changing conditions in near real time. Parties to a contract can reference an oracle to modify payment flows or alter encoded rights and obligations according to newly received information.

In this regard, oracles could be used to determine or update obligations based on the subjective judgment of certain individuals. In this way, parties can rely on "the deterministic and guaranteed execution of smart contracts for objective promises that are readily translatable into code."[32] At the same time, they can choose a human oracle to assess promises that cannot easily be encoded into a smart contract, either because they (1) are too ambiguous, or (2) require a subjective assessment of real-world events.[33]

Despite the benefit of using oracles, using them introduces a potential "point of failure." For example, an oracle might provide erroneous data or simply go out of business.[34] Therefore, parties to smart contracts should be vigilant when choosing their oracles; maybe they should choose more than one substitute to ensure that there will always be an oracle available when needed.

Section (E): Smart Contract Disputes are Inevitable?

Some technologists had proclaimed that smart contracts will avoid disputes altogether on the basis that the parties' bargain is automatically implemented in a decentralized manner, when the conditions

31 Michael del Castillo, *Lawyers Be DAMNed: Andreas Antonopoulos Takes Aim at Arbitration with DAO Proposal*, CoinDesk, May 26, 2016, http://www.coindesk.com/damned-dao-andreas-antonopoulos-third-key. See also Michael Abramowicz, *Cryptocurrency-Based Law*, Arizona Law Review 58 (2016): 359–420 (explaining how blockchains could help facilitate peer-to-peer arbitration, which could lower transaction costs of commercial relationships and increase trust between parties).
32 Ibid.
33 Ibid.
34 Stuart D. Levi and Alex B. Lipton, *An Introduction to Smart Contracts and Their Potential and Inherent Limitations*, Skadden, Arps, Slate, Meagher & Flom LLP and Affiliates, May 7, 2018. Available at https://www.skadden.com/insights/publications/2018/05/an-introduction-to-smart-contracts

agreed between the parties are satisfied. This view is very much overestimated; it does not take into consideration how disputes generally arise in real life. Self-executing smart contracts and blockchain applications might have the potential to increase the efficiency of dispute resolution dramatically. However, disputes will not disappear altogether.

On the contrary, the nature of the blockchain makes it crucial that any aspects of parties' agreement are anchored within a valid legal framework and that the parties identify at the outset the applicable dispute resolution mechanism.[35] Further, smart contracts' disputes would most likely take the form of cross-border disputes because trade is a cross-border activity. Therefore, legal advice on the applicability and enforceability of smart contracts based on the legal framework of each participating jurisdiction will be required beforehand. In this regard, we can identify at least five main potential disputes that could arise in the realm of smart contracts as follows:

I. Is the Smart Contract legally binding? [36]

In most jurisdictions, a contract would only be valid if it is entered into by a person with legal capacity to do so. The fact that pseudonymous parties can enter into smart contracts would make it impossible to validate whether they have the capacity to perform the obligations under such contracts or not. Some civil law jurisdictions lay down some legal requirements (i.e., writing and signing requirements) for the formation of a legally binding contract.

II. Coding limitations as mentioned above might cause unexpected performance issues. [37]

35 Craig Tevendale and Charlie Morgan, *Blockchain and Smart Contracts: novel opportunities for improving efficiency in contract execution and dispute resolution*, Herbert Smith Freehills, Inside Arbitration - Issue 5, February 2018.
36 Available at https://medium.com/humanizing-the-singularity/internet-of-agreements-conference-legal-panel-dadfd7e7ac52; and http://www.nortonrosefulbright.com/knowledge/publications/157162/arbitrating-smart-contract-disputes
37 Ibid.

III. Parties might want to terminate a Smart Contract on the grounds of **misrepresentation, mistake or duress or fraud**.[38] Also conflicts regarding the **definition, interpretation, and general framework** of smart contracts might arise.[39]

IV. **Subsequent changes of law or regulations** might make the performance of the Smart Contract illegal.[40]

V. Smart Contracts might perform on the basis of an **inaccurate data feed** (i.e., an error by the oracle).[41]

Section (F): Is Arbitration the Favourable Dispute Resolution Mechanism for Smart Contract Disputes?

The key features that make arbitration the optimal dispute resolution mechanism for smart contract disputes are arguably the flexibility of arbitral proceedings and the straightforward enforcement of arbitral awards under the New York Convention (Currently there are 159 jurisdictions which are contracting parties). Arbitration could contribute to solving the following issues:[42]

I. Resolving Uncertainty over Jurisdiction & Governing Law.

As smart contracts operate via distributed nodes, it might be difficult to determine the applicable law and the concerned jurisdiction; especially that most of smart contract disputes will take the form of

38 Ibid.
39 Gauthier Vannieuwenhuyse, *Arbitration and New Technologies: Mutual Benefits*, in Maxi Scherer (ed), Journal of International Arbitration, (Volume 35 Issue 1) pp. 119 - 130
40 Available at https://medium.com/humanizing-the-singularity/internet-of-agreements-conference-legal-panel-dadfd7e7ac52; and http://www.nortonrosefulbright.com/knowledge/publications/157162/arbitrating-smart-contract-disputes
41 Ibid.
42 Available at: http://www.nortonrosefulbright.com/knowledge/publications/157162/arbitrating-smart-contract-disputes

cross-border disputes which will usually introduce conflict of laws issues that are extremely challenging to deal with.

II. Protecting Confidential Information.

Some smart contract disputes are likely to involve evidence about proprietary software and/or hardware. The fact that parties can agree to arbitration to make their disputes confidential will enable the parties to limit their exposure and have their confidential information become public.

III. Having a Tribunal with Specialist Technical Knowledge.

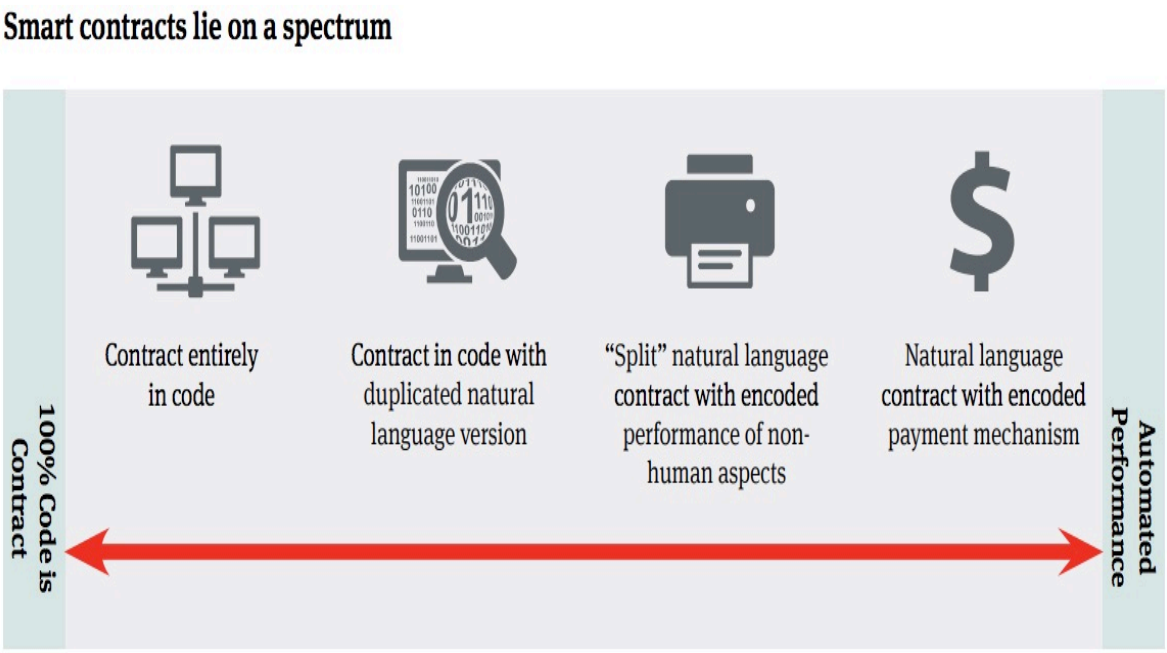
Some smart contract disputes will be fairly vanilla contract law disputes, but others will be of a highly technical nature, for example, where the code does not operate as expected or a technical bug takes place. The courts in many jurisdictions are experienced at dealing with technical issues quickly, but the parties to a smart contract can agree to an arbitration clause which enables them to appoint someone, for example, with an understanding of coding and smart contracts on a certain blockchain (i.e., Ethereum).

IV. Ease of World-Wide Recognition and Enforcement.

Arbitration offers parties the potential to agree to flexible procedures that might help overcome the challenges presented by smart contracts. In addition, the fact that 159 jurisdictions have adopted the New York Convention facilities the process of recognition and enforcement of any arbitral award resulting from a smart contract dispute on a global basis.

Section (G): Main Issues to Consider when Choosing Arbitration to Resolve Smart Contracts Disputes

I. The Form of Smart Contract:[43]



In some jurisdictions, there might be legal risks with having the smart contract entirely in code language. Accordingly, we advise parties to have a hybrid version of smart contract (sometimes called "**Ricardian Contract**") whereby there is a text-based version of the same force in addition to the encrypted coded-language smart contract. Further, the New York Convention requires an agreement to arbitrate to be in writing.[44] In addition, the New York Convention requires an agreement to arbitrate to be signed unless it's in the form of exchange of letters or telegrams.[45] The definition of "an agreement in writing" and "signing" is interpreted differently across the various jurisdictions. It's difficult to predict whether a smart contract encrypted in code would satisfy these requirements

[43] Available at http://www.nortonrosefulbright.com/files/r3-and-norton-rose-fulbright-white-paper-full-report-144581.pdf
[44] See article II of the New York Convention (1958).
[45] Ibid.

beforehand. Therefore, smart contracts run the risk of not being enforced under the New York Convention, unless they have an equivalent traditional word-format contract signed by both parties.

II. The Seat of Arbitration:

Parties to smart contracts should prioritize their choice of the seat of arbitration. A seat of arbitration is not equivalent to a venue for conducting the arbitral hearings. A seat, and a venue are two different things and they are not necessarily the same. In essence, a seat of arbitration underpins the legal framework controlling all legal aspects of the arbitral process. The seat of the arbitration will normally determine the law applicable to the procedure of the arbitration as well as the involvement/ intervention, as appropriate, which the courts of the seat, will have.[46] Also, the seat of arbitration will determine the arbitrability of the subject matter of the dispute and the extent to which the local courts will involve themselves in the arbitral process.

Unfortunately, some jurisdictions are not "arbitration-friendly" as they have laws which restrict party autonomy, for example, by allowing the courts to intervene extensively in the arbitral process.[47] On the other hand, some jurisdictions' laws are relatively "arbitration-friendly" and allow the parties a high degree of procedural autonomy.[48] Whether an arbitral award may be challenged will be determined according to the seat of the arbitration. Further, the extent to which judicial review is available to parties will be dependent on the law of the seat of arbitration.[49] Also, the law of the seat of the arbitration will govern the extent to which an award is considered final. In this regard, it is important to consider a myriad of questions, particularly when deciding upon the seat, including, how the local

46 Available at: https://www.clydeco.com/insight/article/the-seat-of-arbitration-why-is-it-so-important
47 Ibid.
48 Ibid.
49 Ibid.

arbitration law of the seat operates, whether the local courts are "arbitration-friendly,[50] and whether the seat acknowledges the legal binding effects of smart contracts.

III. The Validity and Arbitrability of the Subject-Matter of the Smart Contract:

Before entering into a smart contract, the parties should be aware of the identity of the subject matter of their contract. They should try to investigate whether such a subject matter is valid under the law of the seat of arbitration and also under the law applicable to the merits. In this regard, the parties should also ensure that the subject matter of their smart contract is arbitrable under the law of the seat of arbitration. Failing to inquire about the validity and the arbitrability of the subject matter of the smart contract could deem the arbitration process entirely useless.

IV. The Capacity of the Parties to Enter into the Smart Contracts:

Parties to a smart contract must have the legal capacity to enter into such a contract or otherwise it would be considered invalid. Parties should be aware that their capacity is usually determined by the law of each party, rather than the law of the seat of arbitration or any other law. Therefore, if one of the parties comes from a jurisdiction that does not recognize smart contract, this might affect such a party's ability to enter into the contract. Further, it might serve such a party as a legal loophole to evade its obligations under the smart contract in the future.

V. The Law Applicable to the Merits of the Dispute:

The parties to smart contracts should choose the same jurisdiction for the seat of arbitration and the law applicable to the merits of the dispute. In this regard, jurisdictions such as Arizona, Tennessee, and Delaware are currently considered the friendliest jurisdictions for smart contracts.

50 Ibid.

VI. The Number of Arbitrators:

The parties in international arbitration are usually allowed to choose their arbitrators. The norm is that each party chooses one arbitrator and then both parties or the selected arbitrators, as the case may be, will choose the chair of the arbitral tribunal. The parties should try and avoid choosing an even number of arbitrators as this could be considered to be in violation of various arbitration laws around the world. Also, the parties should not try to choose a number of arbitrators more than three arbitrators or otherwise they might run afoul of the law of the seat of arbitration.

VII. The Technical Qualifications of the Arbitrators:

Parties should try to choose arbitrators who possess the technical knowledge to adjudicate the smart contracts disputes, especially if the dispute is concerning a technical bug for example. This will save the parties time and money when they proceed with arbitration and will enable them to benefit from one of the most important benefits of arbitration.

VIII. The Confidentiality of the Smart Contract Disputes:

Parties should be aware that arbitration is not confidential by default. Therefore, they should provide explicitly for the confidentiality of their dispute under the smart contract. Otherwise, they might run the risk of exposing their confidential information to the public.

Section (H): Survey of Blockchain-Arbitration White Papers: Insightful Remarks

The author surveyed several white papers prepared by the tech community as Blockchain-based arbitration solutions for smart contracts' disputes. The author has carefully selected the white papers included in this survey; so, this is not an exhaustive survey by any means of all the white papers promoting blockchain-based arbitration services. Further, all of the surveyed projects in this paper

except for **Aragon** and **Mattereum** provide principally external arbitration services to supplement the ecosystem of smart contracts running on the Ethereum blockchain. For instance, projects like BitCad[51] were not included in the survey as they were at very early stages and did not provide enough details for their vision of the dispute resolution services.

The author tries to assess how far the tech community is taking into consideration all the potential legal dilemmas associated with arbitrating smart contracts' disputes. Before delving into the details of this survey, we should provide our initial impression of the initiatives proposed by the tech community. In principle, this survey shows that there is a wide gap between the international arbitration community and the blockchain tech community. In other words, the blockchain tech community has not developed a single project that analyzes thoroughly all the risks associated with using the international arbitration mechanism for smart contracts dispute resolution. Therefore, the tech community needs to develop their models exponentially to accumulate enough experience in the field of arbitration of smart contracts, if/when the rate of smart contract dispute raises to a level where it's profitable enough to engage in the field of arbitration of smart contracts.[52]

I. Major Red Flags

1. The Seat of Arbitration and the Applicable Law: This issue might one of the first things that come to mind when dealing with international arbitration, namely, determining the seat of arbitration. This issue is usually crucial because it has so many legal implications ranging from determining the applicable procedural law to being the exclusive forum for annulment proceedings of any arbitral decision or award issued within the seat. The issue of choosing the applicable law is as important as

51 Available at: https://bitcad.io/
52 See the "**Preliminary Statistics of Potential Smart Contract Disputes**" Section mentioned above.

determining the seat of arbitration. However, only one project decided to select the seat of arbitration and the applicable law for its arbitration services.

2. **Arbitrability**: Smart contract disputes can be of various categories; this means that we need to determine whether any of such disputes would be arbitrable under the chosen applicable law. Only one project selected the seat of arbitration and the applicable law for its arbitration services, it's not surprising that none of the projects has considered this issue despite its significance.

3. **Code Language v. Natural Language**: Only one project contemplates the legal risks associated with the code language of smart contracts, and tries to handle this issue by introducing the concept of "**Ricardian contract**."

4. **Formal Requirements of the New York Convention**: Although 4 out of 6 projects acknowledge the existence of the New York convention, only one project contemplated the legal risks associated with the formal requirements of arbitration agreements and arbitral awards under the New York convention.

5. **The Capacity of the Parties and the Arbitrators**: Arbitration contracts usually take the form of arbitral clauses embedded in the main contract in traditional arbitration. In this regard, the issue of the capacity of the parties to enter into the smart arbitration contract would be one of the first issues that would need to be dealt with. However, it seems that only two projects have expressly or impliedly dealt with this issue. This is also the same percentage of projects that have dealt with validating the capacity of the arbitrators.

6. **Confidentiality**: The majority of arbitration practitioners think that arbitration is confidential by default. However, this is an entirely mistaken belief. Therefore, the projects should provide expressly for the confidentiality of the smart contract disputes to resolve this issue. In this regard, only 50% of the projects handled this matter.

7. **Availability of Annulment Proceedings and Penalizing the Arbitrators**: Arbitration is usually in the form of one phase, whereby the merits cannot be reviewed again by any court whatsoever. [53] Despite this, two projects allow for appeal process of arbitration. In addition, the same two projects foresee that there is a right and a wrong answer. In this regard, they penalize the arbitrators whose awards get annulled in the following stage.

II. Secondary Red Flags

1. **Conflict of Interests of Arbitrators**: There is only one project that expressly indicates its intention to handle the conflict of interest of arbitrators.

2. **Legal Qualification of Arbitrators**: The issue of validating the legal qualifications of the arbitrators might be crucial depending on the concerned jurisdiction (i.e., China). This issue was spotted by 50% of the projects.

3. **Arbitral Rules**: Two projects decided to adapt their rules to the UNCITRAL arbitration rules. However, neither explained the reason for such a choice.

53 There is an Exception for example under English Law allowing for appealing the Merits of the Arbitration provided it relates to a point on English Law, and the Court provides its leave for such an appeal.

4. **Arbitral Institutions**: It seems that all the projects except for only one would act as arbitral institutions.

5. **Number of Arbitrators**: One project decided that the default number of arbitrators should be three, while another decided it should be one. A third project proposes a minimum of 5 arbitrators in the first phase, and then nine arbitrators in the supreme court phase. In addition, there will be a mechanism where all judges available on the network can participate in the arbitration. The rest of the projects do not chime in on this issue.

6. **Time Limit of the Arbitral Process**: Only one project expressly refrained from strictly limiting the time limit for the arbitral process.

ANNEX

Rating Checklist for Blockchain-Arbitration White Papers [54]

I. **Does it Determine the Legal Risks associated with having the Smart Contract Encrypted in a Code Language Rather Than a Nature Language (i.e., English)?** [55]

II. **Does it Determine the Legal Risks of the Form of the Smart Arbitration Contracts/Awards: Writing & Signature Requirements under the New York Convention?** [56]

III. **Does it Take into Consideration the Potential Conflict of Interests of the Arbitrators?**

IV. **Does it Validate the Capacity of the Parties to Enter into the Smart Arbitration Contracts / Initiate the Arbitration Proceedings?**

V. **Does it Validate the Capacity of the Arbitrators to Adjudicate the Arbitration Process?**

VI. **Does it Validate the Technical/Legal Qualifications of the Arbitrators?** [57]

VII. **Does it Determine the Seat of Arbitration?** [58]

VIII. **Does it Allow the Parties to Appoint the Arbitrators?**

IX. **Does it Determine the Applicable Laws to the Smart Contract Disputes: (a) Procedural and Evidentiary Matters; (b) The Arbitration Contract; (c) Substantive Matters?**

X. **Does it Take into Consideration the Arbitrability of the Subject-Matter of the Smart Contract Disputes?**

XI. **Does it Determine the Telecommunication Devices used in the Arbitration?**

XII. **Does it Determine the Fees of the Arbitration Proceedings?**

---

54 For further questions, See Below "**Extra Sophisticated Rating Checklist**." The Rating Metric for the Blockchain White Papers operates as follows: Projects Score Points when the Answer is (**Yes**) for **Blue** Questions and (**No**) for **Red** Questions.

55 There is a notion called "**Ricardian Contract**" which could prove to be a success in dealing with the Language Requirements of Smart Contracts. For Reference, See http://iang.org/papers/ricardian_contract.html; https://en.wikipedia.org/wiki/Ricardian_contract; http://iang.org/papers/intersection_ricardian_smart.html

56 Whereas some jurisdictions interpret such requirements broadly while other jurisdictions are very strict.

57 This may be required under some Jurisdictions (i.e., China)

58 It Could be a Pure Legal Fiction where Juris choose the most favorable jurisdiction for Smart Contracts & Arbitration.

XIII. Does it Take into Consideration Any International Arbitration Conventions (especially, the New York Convention) or Arbitral Institutional Rules?

XIV. Does it Take into Consideration the Confidentiality/Privacy Issues of the Smart Contract Disputes?

XV. Does it Propose a Multi-Tiered Dispute Resolution Mechanism (i.e., Non-Binding Negotiation, then Non-Binding Mediation, then Binding Arbitration)?

XVI. Does it Make a Disclaimer for the Annulment/Enforcement Risks of Smart Contract Disputes Arbitration in some Jurisdictions?

XVII. Does it Draft its Own Arbitration Rules from Scratch/Adapt Existing Arbitration Rules?

XVIII. Does it Perform the Role of an Arbitral Institution?

XIX. Does it Refrain from Allowing for the Choice of More Than 3 Arbitrators?[59]

XX. Does it Refrain from Penalizing the Arbitrators for reaching a Different Result than the Majority?

XXI. Does it Refrain from Strictly Limiting the Time for Issuing the Binding Arbitral Award?

XXII. Does it Refrain from Providing for the Annulment Option of the Binding Arbitral Award?[60]

59 There is also an issue with the number of Arbitrators chosen in some Jurisdictions.

60 I do not mean here a Multi-tiered dispute resolution mechanism: For instance, Non-Binding Negotiation Then Non-Binding Mediation Then Binding Arbitration.

Rating Metric for Blockchain-Arbitration White Papers

Issue/Project	Juris[61]	Kleros[62]	Cryptonomica[63]	CodeLegit[64]	Mattereum[65]	Aragon[66]
I	☐	☐	☐	☐	☒[67]	☐
II	☐	☐	☒	☐	☐	☐
III	☒	☐	☐	☐	☐	☐
IV	☐	☐	☒	☐	☐	☒[68]
V	☒	☐	☒	☐	☐	☐
VI	☒	☐	☒	☒	☐	☐
VII	☐	☐	☒[69]	☐	☐	☐
VIII	☒	☒	☐	☒	☒	☒[70]
IX	☐	☐	☒[71]	☐	☐	☐
X[72]	☐	☐	☐	☐	☐	☐
XI	☐	☐	☒	☒[73]	☐	☐
XII	☒	☐	☒	☒	☐	☒

61 **January 2018**: https://jurisproject.io/ ; The answers here are based on both the White Paper of **Juris** and an Interview with the Co-Founder of this Project, Mr. **Adam J. Kerpelman**, the CEO of **Juris.**
62 **January 2018**: https://kleros.io/
63 **November 2017**: https://cryptonomica.net/#/
64 **July 2017:** http://codelegit.com/
65 **Unspecified Month 2017:** https://mattereum.com/; http://internetofagreements.com/.
66 **April 2017**: https://aragon.one/
67 **Mattereum** introduces the concept of a "**Ricardian Contract**" to deal with this issue.
68 **Aragon** proposes that "only applicants that are among the defendant's organization shareholders will be allowed for this petition," **Aragon White Paper**, Page 28.
69 **Cryptonomica** selects London, the U.K as the seat of arbitration for its reputation as an International Arbitration Hub.
70 **Aragon** proposes 2 Arbitration Mechanisms, namely: (1) **Human Judges**: where a selected number of judges participate in the arbitration; and (2) **Prediction Market:** where all available judges on the Aragon Network participate in the Arbitration. Aragon White Paper, Page 28-9.
71 **Cryptonomica** chooses the U.K Law as the applicable law, without specifying the various potential applicable laws that could be applied to Smart Contracts Disputes.
72 Despite the Critical Importance of this issue, none of the Surveyed White Papers has paid any attention to it.
73 **CodeLegit** proposes an automated Arbitral Process where a Blockchain Arbitration Library would be running the Arbitral Process.

XIII	☒	☐	☒[74]	☐	☒	☐
XIV	☐	☒	☐	☒[75]	☐	☒
XV	☒	☐	☐	☐	☐	☐
XVI	☐	☐	☐	☐	☐	☐
XVII	☒[76]	☐	☐	☒[77]	☒[78]	☒[79]
XVIII	☒	☒	☒	☒	☐	☒
XIX	☒[80]	☐	☐	☒[81]	☐	☐[82]
XX	☒	☐[83]	☐	☐	☐	☐[84]
XXI	☒	☒	☐	☐	☐	☐
XXII	☒	☐	☐	☒	☐	☐
Score	14	4	10	8	N/A[85]	6

74 **Cryptonomica** is the only project that recognizes the existence of the Apostille Treaty by The Hague (1961) in addition to acknowledging the Existence of the New York Convention.
75 **CodeLegit** seeks to preserve the Confidentiality of the Arbitral Process through hashing all relevant emails.
76 **Juris** seeks to adapt its Rules to the UNCITRAL Arbitration Rules.
77 **CodeLegit** seeks to adapt its Rules to the UNCITRAL Arbitration Rules.
78 It seems that **Mattereum** will import the Arbitral Rules of Existing Arbitral Institutions.
79 This takes the form of allowing the Aragon Network Token (ANT) holders of voting on the basic rules that will be considered by the Arbitrators.
80 **Juris** proposes a default number of 3 Arbitrators.
81 **CodeLegit** proposes a default number of 1 Arbitrator.
82 **Aragon** proposes a minimum of 5 Arbitrators in the first phase, and then 9 Arbitrators in the Supreme Court Phase; in addition, there will be a mechanism where all Judges available on the Aragon Network can participate in the Arbitration.
83 **Kleros** foresees that there is a Right Answer and a Wrong Answer & Penalizes the Arbitrators who get it wrong. This is sort of an application of the Game Theory in Arbitration.
84 **Aragon** also foresees that there is a Right Answer and a Wrong Answer & Penalizes the Arbitrators who get it wrong. This is sort of an application of the Game Theory in Arbitration.
85 We cannot precisely score **Mattereum** because the terms of the contemplated association with Arbitral Institutions are not available for the Public. In any case, **Mattereum** seems to be a Promising Project because (1) it introduces the Concept of "**Ricardian Contracts**" & (2) Acknowledges the Existence of the New York Convention.