DISPUTE RESOLUTION JOURNAL®

AMERICAN ARBITRATION ASSOCIATION®
INTERNATIONAL CENTRE FOR DISPUTE RESOLUTION®
AAA Mediation.org

OCTOBER 2020 | VOLUME 74 | NUMBER 4

Arbitrating Disputes Involving Blockchains, Smart Contracts, and Smart Legal Contracts

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

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, New York and Rumson, New Jersey. 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; a Chartered Arbitrator and Fellow of the Chartered Institute of Arbitrators, and Chair Emeritus and Co-Founder of the New York Branch of CIArb; and has been recognized by the Silicon Valley Arbitration & Mediation Center (SVAMC) as a leading technology arbitrator and mediator on their "Tech List." He holds a LL.M. (Trade Regulation) from NYU School of Law, a J.D. from Duquesne University, and an M.S. in Electrical Engineering and a B.S. 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 and a B.A. in computer science. Before practicing law, she worked as a computer scientist. She serves as a neutral for the AAA, ICDR, and CPR; is a Fellow and Chair of the North American Branch of the Chartered Institute of Arbitrators (CIArb); and has also 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 with, was described in Stuart Haber et al., *How to Time-Stamp a Digital Document*, 3 J. of Cryptology, no. 2, Jan. 1991, at 99-111. In 1992, the system was expanded to allow several document certificates to

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 of 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."

I. 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 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 each party trusts. Whether that intermediary is 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 passage from the *MIT Technol-*

be collected into one block. David Bayer et al, *Improving the Efficiency and Reliability of Digital Time-Stamping*, 2 Sequences, Mar. 1992, at 329-34. What appears to be the first conceptualization of blockchain was made by a person or persons known as Satoshi Nakamoto in 2008—though the exact identity of Nakamoto remains a mystery in the cryptographic field—when a paper was published under that name describing the implementation behind the cryptocurrency Bitcoin. Nakamoto incorporated hash methodology to timestamp blocks 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* (2008), https://bitcoin.org/bitcoin.pdf; *see also* Arvind Narayanan et al., *Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction* (Princeton Univ. Press 2016).

ogy Review demonstrates how the need for trust drove the historical use of ledgers, double-entry accounting, and ultimately blockchains:

> 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 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. . . .
>
> The real promise of blockchain technology . . . is that it could drastically reduce the cost of trust by means of a radical, de-

centralized approach to accounting—and, by extension, create a new way to structure economic organizations.

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 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 . . . .[2]

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 just 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 are 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's 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 Contracts

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

---

[2]Michael J. Casey & Paul Vigna, *In blockchain we trust*, MIT Tech. R. (Apr. 9, 2018) https://www.technologyreview.com/s/610781/in-blockchain-we-trust/.

[3]*Id.*

[4]https://digitalchamber.org/initiatives/smart-contracts-alliance/.

[5]https://digitalchamber.org/.

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 Contracts

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 Contracts Alliance 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

[6]Mark M. Higgins, *Blockchain in Energy: Smart Legal Contracts on the Rise*, Nat'l L. Rev. (July 26, 2019), https://www.natlawreview.com/article/blockchain-energy-smart-legal-contracts-rise.

[7]*Id.*

[8]Oracles retrieve and verify external data for blockchains and smart contracts.

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."[10]

In that regard, the Accord Project, a nonprofit open-source consortium aimed at transforming contract management and contract automation, is developing an open, standardized format for Smart Legal Contracts[11] 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.[12] 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 for anyone, through use of those tools and the ecosystem, can draft Smart Legal Contracts in a

[9]Smart Contracts Alliance, *Smart Contracts: Is the Law Ready?*, Chamber of Digital Commerce, 12 (2018), https://digitalchamber.org/download/9420/ [hereinafter: "*Smart Contracts: Is the Law Ready?*"].

[10]Accord Project, Overview, https://docs.accordproject.org/docs/accordproject.html.

[11]Accord Project, 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, *Clyde & Co launches connected parametric insurance contract*, Clyde & Co. Newsletter (May 15, 2019). In the U.S., "Latham & Watkins has teamed up with ConsenSys to develop a Smart Legal Contract that automates convertible note agreements. . . . [T]his 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, *supra* note 6.

[12]Accord Project, https://docs.accordproject.org/.

standardized neutral, technology agnostic format once—and then use and reuse it, as often as desired, across a variety of supported technologies.[13]

The Global Legal Blockchain Consortium (GLBC) is another nonprofit 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.[14] 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 alternative dispute resolution community of interest to explore "on-chain" and "off-chain" arbitration of blockchain disputes.

3. Ricardian Contracts

The Ricardian contract, being similar to a Smart Legal Contract and conceived of by financial cryptographer Ian Grigg, is a contract "represented both in plain text and in digital code[,]" digitally signed to provide it with all the elements of a standard legal contract.[15] 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."[16] *Forbes* described the Ricardian Contract as "a smarter and more useful digital contract."[17] There are obvious efficiency and cost advantages to Smart Legal Contracts and Ricardian contracts. Not

---

[13]Accord Project, *supra* note 11.

[14]Global Legal Blockchain Consortium, https://legalconsortium.org/what-is-the-glbc/.

[15]Distributed.com, *Filling in the Missing Piece of Smart Contracts* (Aug. 15, 2018), https://www.nasdaq.com/articles/filling-missing-piece-smart-contracts-2018-08-15.

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

[17]Chao Cheng-Shorland, *Moving Beyond Smart Contracts: What Are The Next Generations Of Blockchain Use Cases?*, Forbes (Dec. 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.

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.[18]

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 Supervisory Control and Data Acquisition (SCADA) system that American power plants use to decide how much power to generate and where to send it. As SCADA can be a huge central point of attack, the U.S. Department 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.[19]

---

[18] Accord Project, *supra* note 10.

[19] Daniel Tkacik, *Securing the Energy Grid with Blockchains*, Carnegie Mellon Eng'g Magazine, Fall 2019, at 28.

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 years, a number of multistate 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 specific producers, facilities, or regions for appropriate remediation.[20] 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 the produce or sells it 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.[21] 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.

---

[20]Centers for Disease Control, *Outbreak of E. coli Infections Linked to Romaine Lettuce—Final Update* (Jan. 9, 2019), https://www.cdc.gov/ecoli/2018/o157h7-11-18/index.html.

[21]IBM, Transform supply chain transparency with IBM Blockchain, https://www.ibm.com/downloads/cas/1VBZEPYL [hereinafter "IBM Supply Chain White Paper"]; IBM, Who will win the race to blockchain supply chain supremacy?, https://www.ibm.com/blockchain/industries/supply-chain; *see also* Sloane Brakeville et al, *Blockchain basics: Glossary and use cases*, IBM Developer (Aug. 21, 2017), https://developer.ibm.com/tutorials/cl-blockchain-basics-glossary-bluemix-trs/.

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.[22]

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 U.S. The project was authorized under the U.S. Drug Supply Chain Security Act, which increased the FDA's ability to help protect consumers from exposure to counterfeit, stolen, contaminated, or otherwise harmful drugs.[23]

II. 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.

A. A Primer on the Technologies

1. Blockchains and Blockchain Ledgers

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

As shown, the block contains transaction data for a given transaction and its hash value. (As more specific details are irrelevant to this explanation, they have been omitted for simplicity.) 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

[22]IBM Supply Chain White Paper, *supra* note 21; *see also* Chamber of Digital Commerce, *National Action Plan for Blockchain—The Need for a Comprehensive, Coordinated, Pro-Growth Approach to Developing Blockchain Technology in the United States* (Feb. 2019), https://digitalchamber.org/wp-content/uploads/dlm_uploads/2019/02/National-Action-Plan-for-Blockchain1.pdf.

[23]Edward Pearcey, *U.S. border agency tests IP blockchain solution*, World Intell. Prop. Rev. (Feb. 3, 2020), https://www.worldipreview.com/news/us-border-agency-tests-ip-blockchain-solution-19392.

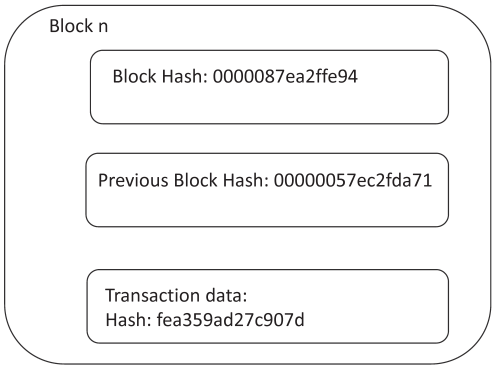

Figure 1. Typical blockchain block

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 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[24]) 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;

[24] *See, e.g.*, XORBIN.com, SHA-256 hash calculator, https://xorbin.com/tools/sha256-hash-calculator.

- 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 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").[25]

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.

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

Conventional business networks for recording transactions, simplistically illustrated by that depicted in Figure 3, rely on each party, A-D, to write transaction data into its own database (containing respective Ledgers A-D) and communicate 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,

[25]Manav Gupta, *Blockchain for dummies, IBM Limited Edition*, 13-14 (John Wiley & Sons 2017), http://gunkelweb.com/coms465/texts/ibm_blockchain.pdf; *see also* Anastasiia Lastovetska, *Blockchain Architecture Basics: Components, Structure, Benefits & Creation*, MLSDev (Jan. 31, 2019), https://mlsdev.com/blog/156-how-to-build-your-own-blockchain-architecture.

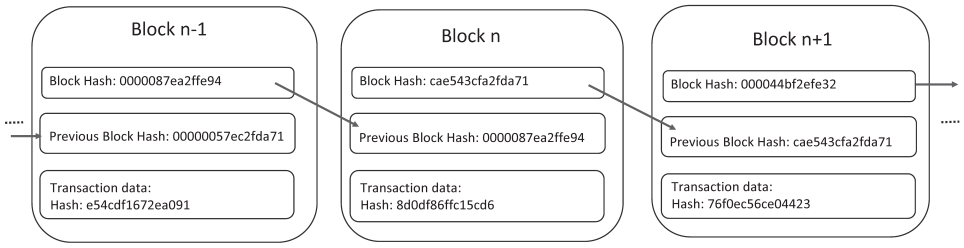

Figure 2. Interconnected blockchain blocks

cyberattack, or just a simple human mistake, incorrect transaction data will propagate to and adversely affect transaction data stored in all the other ledgers.

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.[26] As will be described in further detail below, each node writes all transactions, once validated, into its replica of the blockchain, thus the

[26]Casey & Vigna, *supra* note 2.

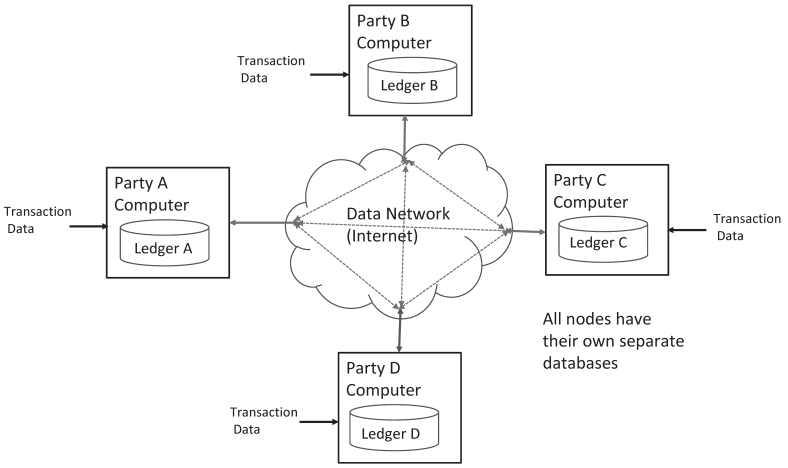

Figure 3. Conventional business network

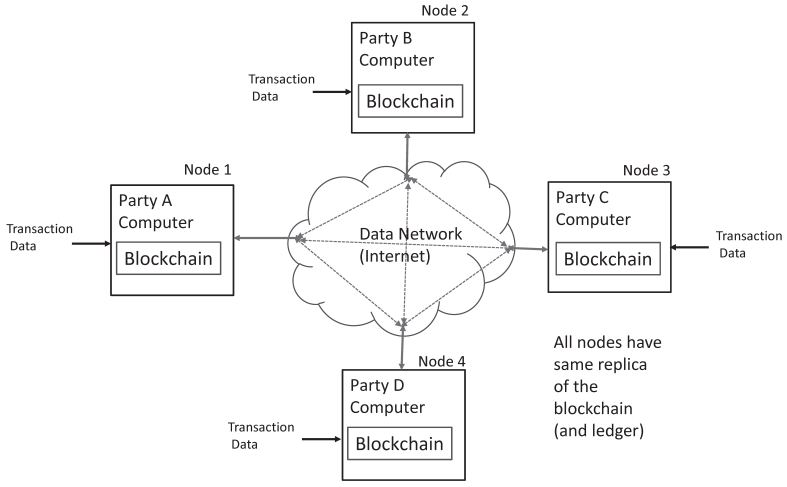

Figure 4. Four-node blockchain network

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 either physically or, more likely, in a virtualized form. 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 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.[27]

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

[27]Microsoft Azure, Blockchain, https://azure.microsoft.com/en-us/solutions/blockchain/; IBM, Blockchain, https://www.ibm.com/blockchain.

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 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:

1. 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.

2. 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.

3. 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.[28]

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

[28]Gupta, *supra* note 25, at 15.

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 (parties) that are transacting together, such as a supply chain, which requires a common, secure, immutable record-keeping system but where those organizations are otherwise independent of each other and may not fully trust each other.[29]

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.[30] As permissioned

[29]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; *see also* Gupta, *supra* note 25, at 16.

[30]*See, e.g.*, Andrew Tar, *Proof-of-Work Explained*, CoinTelegraph (Jan. 17, 2018), https://cointelegraph.com/explained/proof-of-work-explained.

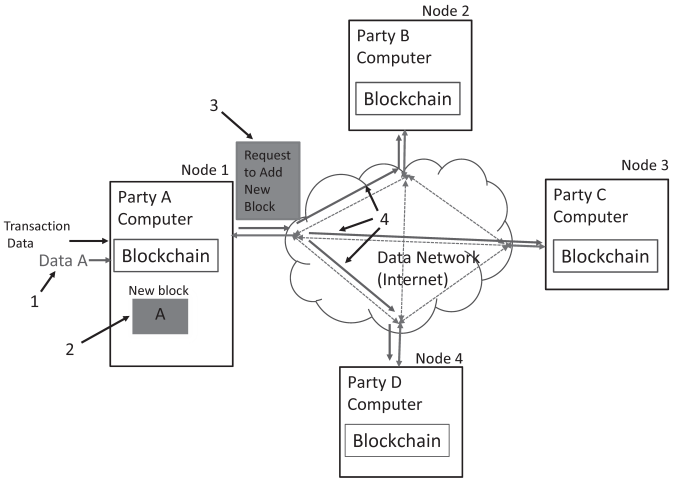

Figure 5. New block generation

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).

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,

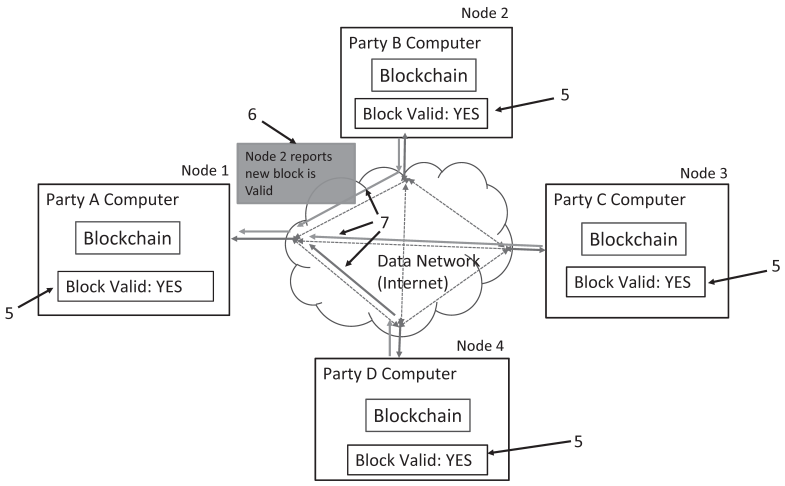

Figure 6. Validity determination

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.

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, 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

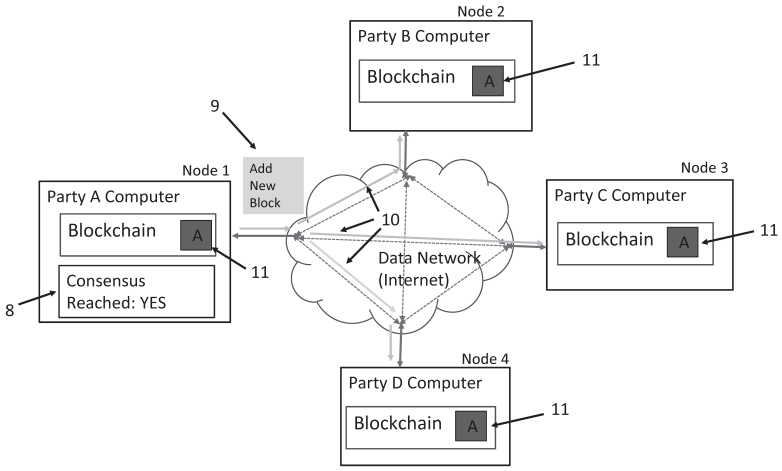

Figure 7. Appending new block to blockchain

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.

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 been illicitly modified, altered, or changed in any way.[31]

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

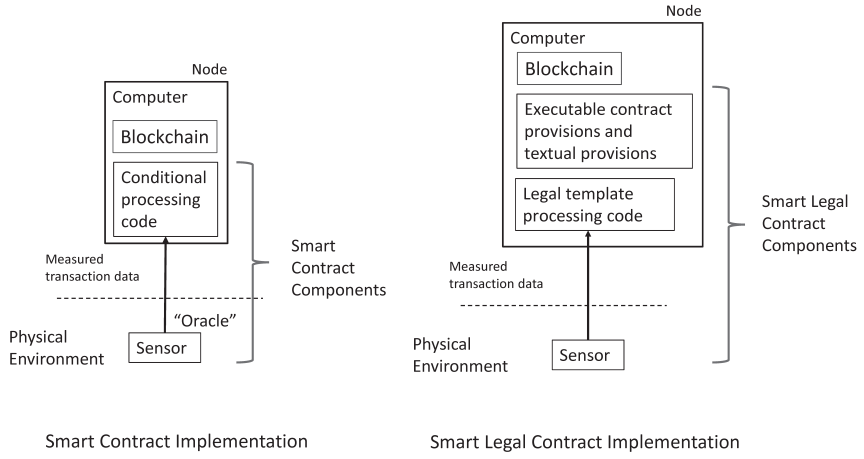

Figure 8. Smart Contract and Smart Legal Contract implementation

2. Smart Contracts and Smart Legal Contracts

Figure 8 depicts, at a very high level, 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.

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.[32] 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

[32]Blockgeeks, What is Ethereum?, https://blockgeeks.com/guides/ethereum/.

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: "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 legally binding contract under common law.[33]

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 to be 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 a framework similar to the one 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

[33] *Smart Contracts: Is the Law Ready?*, *supra* note 9.

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.[34]

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. That message might include the names and addresses of the parties, the sale price to be paid in a domestic currency (e.g., U.S. 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 III to various legal

[34]Accord Project, *supra* note 10.

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

III. 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 to 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 plane crashes killing 346 people.[35] 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.[36]

---

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

[36]Ben Popken & Maura Barrett, *Iowa caucus app was rushed and flawed from the beginning, experts say*, NBC News (Feb. 5, 2020), https://www.nbcnews.com/tech/security/iowa-caucus-app-was-rushed-flawed-beginning-experts-say-n1131216.

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 in 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 the 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

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.[37] While software development has existed for decades, smart contract development platforms were only developed in 2015. Due to the recent development of such platforms, there is a notable absence of developer handbooks relating to smart contracts.[38] 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

---

[37]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/.

[38]Yos Riady, *Best Practices for Smart Contract Development*, (Nov. 10, 2019), https://yos.io/2019/11/10/smart-contract-development-best-practices/.

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 involves "The DAO" exploit, which we will now discuss.

The DAO Incident

Distributed Autonomous Organizations (DAOs) are run by programming code and constitute a collection of Smart Contracts[39] operating independently of any human intervention, as long as funding covers a DAO's survival costs and provides a useful service to its participant base.[40] 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 provide the DAO with operating 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 the person controlled. The specific error in the code was known to The DAO's creators, but it was 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

[39] 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* Blockgeeks, supra note 32.

[40] *Id.*

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.[41] Since an error existed in the code, The DAO sought to renegotiate the terms—notwithstanding the fact that renegotiation is arguably contrary to the fundamental notion of 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, embodying 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 particular conditions are 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 a question of whether predefined conditions have been objectively satisfied or not. Rather, they call for a determination that requires some degree of human subjectivity. Specifically, the parties or an adjudicator may need to assess subjectively

[41] *Id.*

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 others, which collectively form the foundation of contract law.[42] Consequently, these legal agreements, by their very nature, are inappropriate for codification and implementation as smart agreements.

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, straightforward transactions, to anticipate every such contingency that might possibly arise, no matter how small its probability of occurrence. 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 incompleteness can be exploited in a way that results in a better contract for them, *ex-ante*. Renegotiation is a common way that ambiguities are resolved and contractual gaps filled.[43] Parties need some degree of flexibility in resolving contractual incompleteness that avoids locking themselves into rigid commitments and outcomes which they did not anticipate and do not want.[44]

Consequently, for other than relatively simple, completely deterministic transactions, it is quite possible that the code in smart agreements will fail

---

[42]Pietro Ortolani, *The impact of blockchain technologies and Smart Contracts: arbitration and court litigation at the crossroads*, 24 Uniform L. Rev., Issue 2, at 438 (June 2019), https://academic.oup.com/ulr/article/24/2/430/5490658.

[43]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.

[44]Houman B. Shadab, *What Smart Contracts Need to Learn*, Lawbitrage, (Sept. 4, 2014), https://lawbitrage.typepad.com/blog/2014/09/smart-contracts.html.

to reflect some contingencies. Code is not subject to renegotiation. Smart agreements, once they are embodied into code, are fixed. As The DAO incident showed, some smart agreement adherents vociferously advocate that "The Code is Law" (i.e., the ultimate arbiter of a deal it represents—a standalone, self-enforcing agreement not subject to interpretation by outside entities or jurisdictions).[45] 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 with 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.[46] While exploited vulnerabilities have captured the headlines and imaginations, recent academic research reported that, out of 21,270 vulnerable smart contracts, at most

[45]David Siegel, *Understanding the DAO Attack*, Coindesk (June 25, 2016), https://www.coindesk.com/understanding-dao-hack-journalists.

[46]Daniel Perez & Benjamin Livshits, *Smart Contract Vulnerabilities: Does Anyone Care?* (May 17, 2019), https://arxiv.org/pdf/1902.06710.pdf.

only 504 have been subjected to exploits, likely due to the fact that a majority of Ether is held by only a small number of contracts.[47]

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 appears on a Blockchain Ledger can exercise their right to have that 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

[47] *Id.*

10% in 2020.[48] 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

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 initial coin offerings (ICOs) where the ICOs were to be implemented on blockchains.[49] 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

[48]Susan Moore, *Gartner Predicts for the Future of Privacy 2020*, Smarter with Gartner (Jan. 20, 2020), https://www.gartner.com/smarterwithgartner/gartner-predicts-for-the-future-of-privacy-2020/.

[49]*See, e.g.*, *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.*, 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.*, 17-cv-24500-JLK (S.D. Fla.); *Hodges v. Monkey Capital, LLC*, 17-81370 (S.D. Fla.); *Balestra v. ATBCOIN, LLC*, 17-10001 (S.D.N.Y.); *Stormsmedia, LLC v. Giva Watt, Inc.*, 17-00438 (E.D.Wash.); *Davy v. Paragon Coin, Inc.*, 18-00671 (N.D. Cal.). Also, for SEC concerns regarding ICOs, see https://www.sec.gov/ICO.

with jurisdiction: over the parties for *in personam* jurisdiction or over an object 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 "ones" and "zeroes" 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.[50] 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.

---

[50] 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 virtualization software providers, such as 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)[51] and the "Uniform Electronic Transactions Act" (UETA)[52] 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 U.S. 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 interstate commerce.[53] Though ESIGN will preempt any inconsistent state law, it expressly precludes preemption of UETA in any state or territory that enacted the latter.[54] UETA, in contrast to ESIGN, has no consumer notice provision, though certain enacting states have enacted their

[51] 15 U.S.C. § 7001, *et seq.* (2000).

[52] Approved and recommended by the Uniform Law Conference in 1999 for state enactment.

[53] RightSignature, UETA—Uniform Electronic Transactions Act, https://rightsignature.com/legality/ueta-act.

[54] Margo H. K. Tank et al., *A short primer on applicable US eSignature laws* (May 2, 2018), 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/.

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.[55] 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 in smart agreements or for storing certain records (Illinois—May 29, 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).[56]

IV. 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,

[55] *Id.*

[56] Margo H. K. Tank et al, *Blockchain and Digital Assets News and Trends* (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/.

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.[57] 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.[58] 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.[59]

Such platforms may ultimately prove useful in efficiently 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.

As such, an effective practical approach for blockchain administrators may well be to impose a contractual framework onto all their participants to which each participant would assent as a condition for joining the blockchain.

---

[57]Sara Hourani, *The Legal Reality of the Recognition and Enforcement of Cross-Border Blockchain-based Arbitral Awards: Beyond Futuristic Idealism?*, Off the Chain (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.

[58]Clement Lesage et al., *Kleros, Short Paper v. 1.0.7* (Sept. 2019) https://kleros.io/assets/whitepaper.pdf.

[59]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/.

That framework would: specify a certain arbitral forum (e.g., the 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.[60] 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.

A. Protection of Proprietary Information

Protection of proprietary information is not only important to the parties. 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

[60] 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 .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, wherever situated 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.

neutrals an ethical obligation to keep information confidential.[61] 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. The AAA and its international division, the International Centre for Dispute Resolution (ICDR) has, for example, implemented best practice policies, technologies, and procedures to help protect its data and technology resources.[62] 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, it also gives parties involved in the dispute the opportunity to select their arbitrator(s), giving all parties confidence that an equitable solution will be reached.[63]

With the complexity of the underlying technology and likelihood of technical issues, it is 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.

---

[61] *AAA Statement of Ethical Principles*, https://www.adr.org/StatementofEthicalPrinciples.

[62] "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.

[63] P. Jean Baker, *Arbitrators Provide Technical Expertise, Confidentiality*, Corp. Counsel Bus. J. (Jan./Feb. 2020).

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 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.[64] 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 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 and Rule 38 of the AAA Commercial Arbitration Rules. An

[64]Peter L. Michaelson, *Patent Arbitration: It Still Makes Good Sense*, 7 Landslide (publication of the ABA Section of Intellectual Property Law), no. 6, at 46 (July/Aug. 2015).

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.

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.

V. 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 I(B) of this paper, a Ricardian Contract or a smart legal contract, that includes both "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 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*, a 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 and 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 using 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 still likely that a need will 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.[65]

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 must 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

[65] *Smart Contracts: Is the Law Ready?*, *supra* note 9.

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 addresses the issue.[66]

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

---

[66] Sara Hourani, *The Legal Reality of the Recognition and Enforcement of Cross-Border Blockchain-based Arbitral Awards: Beyond Futuristic Idealism?*, Off the Chain (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.

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. The advantages obtainable through institutional administration,[67] often significantly outweigh whatever cost savings an ad hoc process might provide, let alone when in a

[67]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.

complex and time-sensitive proceeding as a blockchain-related arbitration is likely to be.

Conclusion

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 expanding use across 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.