BOT CONTRACTS

Deborah R. Gerhardt & David Thaw*

In this Article, we explain why the transactions commonly known as "smart contracts" are better understood as "bot contracts." Taking an interdisciplinary approach, we show why the "smart contracts" moniker is misdescriptive in two important ways. First, these transactions are automated, not smart. Second, they do not afford parties many enforcement rights and defenses that one expects from common law contractual relationships. To fully understand these transactions, it is important to appreciate how the term "smart contracts" differs from what the technology delivers.

Our review of the technology explains that these transactions have tremendous practical utility in reducing risk and avoiding the uncertainty and expense of seeking judicial enforcement. However, the electronic processes that occur in this category are not smart in the sense of being thoughtful, creative, or even amenable to change. They are programmed to follow preset instructions and execute automatically. Once the conditions for performance under a smart contract occur, performance cannot be stopped. Because these transactions are automated, they lack features and defenses available to those who enter into typical contractual relationships. Common law contracts are sets of promises or obligations that may be enforced by a court. However, once a smart contract is set in motion, no person or court can reverse the transaction. In this way, smart contracts differ fundamentally from traditional contracts because they leave no room for judicial intervention. By design, they evade the risk of what a court may do in fashioning a remedy. Courts have no power to set the transaction aside if it was induced by fraud or if another

[*] Deborah R. Gerhardt is the Reef C. Ivey II Excellence Fund Term Professor of Law at the University of North Carolina, Chapel Hill. David Thaw is an Associate Research Professor of Law, an Assistant Research Professor of Computing & Information at the University of Pittsburgh, an Affiliated Fellow of the Information Society Project at Yale Law School, and Faculty Director and Founder of the CyREN Laboratory. Beginning in 2019, Professor Thaw has been on research leave leading the Obnostic Project, a distributed computational platform based on the fundamental science underlying blockchain technology. Professor Thaw receives no compensation, other than his faculty salary, for his work on this paper, and the paper neither is funded nor influenced by Obnostic LLC or any of its affiliates. The Authors are grateful to Douglas Baird, Michael Malloy, and the participants at KCON 2020 for helpful feedback. We are also grateful to Taylor Townes and Chandler Martin for excellent research assistance.

Electronic copy available at: https://ssrn.com/abstract=3533517

common law defense would, under other circumstances, render the transaction void or voidable. Although the term "smart contract" appears to have taken hold, we propose that these transactions are better thought of as "bot" or "automated" agreements. Reframing these transactions in this way would reset expectations in line with what the technology can deliver. Adopting this more encompassing terminology will send a strong informational signal that avoids misrepresenting the abilities of these agreements by more accurately communicating that they execute automatically and eliminate both the risks and benefits that accompany traditional common law contracts.

INTRODUCTION

The term "smart contracts" seems to have taken on a life of its own. The moniker was coined by a computer scientist to describe software that worked like a vending machine.[1] In the following decades, its meaning has snowballed to the point where the term has picked up power and meaning that extends beyond what the technology offers. To understand these transactions, one must consider how smart contracts differ from common law contracts.

Smart contracts occur when two people tell software to conduct an activity. Unlike a traditional common law contract, in which offer and acceptance of mutual promises occur, entering into a smart contract is like standing with a friend and agreeing to press the button that unlocks your car doors. Once you both press the button, there is no turning back. The software sends a signal. The car will unlock. Pressing the button again can cause the car to lock, but it will not change the first unlocking. Smart contracts work in the same way. Two people may decide to complete a sale if a certain condition (analogous to pressing the lock button) occurs,

[1] Roberto Pardolesi & Antonio Davola, What Is Wrong in the Debate About Smart Contracts 1 (Feb. 21, 2019) (unpublished manuscript), https://ssrn.com/abstract=3339421 or http://dx.doi.org/10.2139/ssrn.3339421.

Electronic copy available at: https://ssrn.com/abstract=3533517

the instructions are sent, and the transaction will perform. Once the triggering conduction happens, formation of the contract cannot be stopped or undone. For this reason, the term "smart contracts" is technically misdescriptive in two ways. The electronic processes that fit in this category are not smart. They do not think, create, or choose. Rather, they are programmed, not to think, but to follow preset instructions. And strictly speaking, they are not contracts. Common law contracts are sets of promises or obligations that may be enforced at law. Once a smart contract is set in motion, no person or court can reverse the transaction. Like electronic door locks, smart contracts can add tremendous value and efficiency by minimizing effort, error, and risk.[2]

In this Article, we illustrate how these automated transactions challenge the foundational notions of contract law and have important implications for parties considering what type of platform to select for their transactions. Part I identifies basic features that separate enforceable contracts from other promises. Part II summarizes the technology that enables private actors to enter smart contracts. Once that foundation is set, it becomes clear that smart contracts lack several features of common law contracts. These features are identified in Part III. Most notably, smart contracts leave no room for judicial intervention. By design, they eliminate the uncertainty of what a court may do in fashioning a remedy. They are, therefore, not the kinds of transactions for which a court can provide a remedy or order the parties to do anything. Courts have no power to set the transaction aside if it is based on fraud or if a common law defense would, under other circumstances, provide a reason to void the transaction. For all of these reasons, the agreements known as "smart contracts" offer a mix of benefits and challenges which are obscured by their common name. Although the term "smart contract" appears to have taken hold, we suggest that these transactions are better thought of as "bot" or "automated" contracts because of the many ways they differ from traditional contracts that come with a host of common law remedies and a set of expectations that automated transactions are designed not to provide.

I. DEFINING FEATURES OF COMMON LAW CONTRACTS

A basic tenet of contract law is that only a subset of mutual decisions and exchanged promises amount to enforceable contracts.[3] The law of contracts defines these boundaries.[4] Contracts that are bargained exchanges between adult actors with the capacity to enter into agreements are protected by contract law.[5] Although they are private agreements, they emanate an aura of authority arising out of the accepted notion that courts will enforce them. Not all promises fall within this boundary. The

[2] Smart contracts purport to bring together the advantages of a self-executing contract with the customization options of traditional common law contractual drafting. Furthermore, they reduce resolution costs by largely removing enforcement from the equation. However, as this Article addresses, these advantages are not always present and not all smart contracts deliver on the full promises of traditional common law contracts.

[3] *See* RESTATEMENT (SECOND) OF CONTRACTS § 1 (AM. L. INST. 1981) ("A contract is a promise or a set of promises for the breach of which the law gives a remedy, or the performance of which the law in some way recognizes as a duty.").

[4] *See generally id.*

[5] *See id.* § 12.

possibility of court intervention to identify and enforce an agreement is an important element in defining this boundary between enforceable contracts and unenforceable promises or agreements. Professor Joseph Perillo begins his contracts treatise with a discussion on the difficulty of defining a contract.[6] But all the competing definitions share a common theme. They embrace the understanding that "[e]very contract involves at least one commitment that has legal consequences."[7] In its very first section, the *Restatement (Second) of Contracts* uses this notion in defining a contract. The *Restatement* provides that "[a] contract is a promise or a set of promises for the breach of which the law gives a remedy, or the performance of which the law in some way recognizes as a duty."[8] The Uniform Commercial Code confirms this basic foundation, noting that even if an agreement is missing a key term, it still may be considered a contract if "there is a reasonably certain basis for giving an appropriate remedy."[9] In other words, an agreement will be deemed a contract only if, by its terms, a neutral arbiter can identify and award a remedy.

The common law tradition developed the requirement that a set of promises rises to the level of a contract only if a court can enforce it. For example, if a set of promises is not sufficiently definite for a court to intervene and provide a remedy, it will not be deemed an enforceable contract. Professor Samuel Williston explains that "[i]t is a necessary requirement that an agreement, in order to be binding, must be sufficiently definite to enable the courts to give it an exact meaning."[10] State courts similarly rely on the possibility of judicial enforcement as a metric for determining whether a contract exists. For example, the Alabama Supreme Court opined that "any contract must express all terms essential to the transaction with definiteness *sufficient to enable a court to enforce the parties' agreement*."[11] A court's ability to identify contractual obligations and fashion a remedy is essential to determining whether expressions and conduct fit within the sets of promises the law recognizes as contracts.

This basic existential notion is not unique to the United States. The United Nations Convention on Contracts for the Sale of Goods indicates that it is not intended to address whether an agreement constitutes a valid contract.[12]

[6] JOSEPH M. PERILLO, CALAMARI AND PERILLO ON CONTRACTS § 1.1, at 1 (7th ed. 2014) ("No entirely satisfactory definition of the term 'contract' has ever been devised. The difficulty of definition arises from the diversity of the expressions of assent which may properly be denominated 'contracts' and from the various perspectives from which their formation and consequences may be viewed.").

[7] *Id.*

[8] RESTATEMENT (SECOND) OF CONTRACTS § 1 (AM. L. INST. 1981).

[9] U.C.C. § 2-204(3) (AM. L. INST. & UNIF. L. COMM'N 1977) ("Even though one or more terms are left open a contract for sale does not fail for indefiniteness if the parties have intended to make a contract and there is a reasonably certain basis for giving an appropriate remedy.").

[10] 1 SAMUEL WILLISTON, A TREATISE ON THE LAW OF CONTRACTS § 4:21, at 634 (Richard A. Lord ed., 4th ed. 2007).

[11] Capmark Bank v. RGR, LLC, 81 So. 3d 1258, 1268 (Ala. 2011) (emphasis added) (quoting Macon Cty. Greyhound Park v. Knowles, 39 So. 3d 100, 108 (Ala. 2009)).

[12] United Nations Convention on Contracts for the International Sale of Goods art. 4(a), Apr. 11, 1980, 1489 U.N.T.S. 3 ("This Convention governs only the formation of

Electronic copy available at: https://ssrn.com/abstract=3533517

Accordingly, the entire convention is premised on the notion that one or more tribunals may enforce agreements within its scope.[13]

II. HOW SMART CONTRACTS WORK

Evaluating the ability of smart contracts to implement the requirements of common law contract doctrine first requires some basic understanding of blockchain technologies, their history, and the development of the technologies currently referred to as "smart contracts." This is particularly important because we are considering an emerging technology, the limitations (and abilities) of which continue to evolve over time. This Part begins with some "Blockchain Basics," followed by a contextual overview of the history of blockchain technologies and the rise of smart contracts. It concludes by contextualizing the current abilities of smart contracts to fulfill contract doctrinal requirements. In particular, we focus on two apparent deficiencies: the implied right of parties to breach and the legal requirement of capacity.

A. Blockchain Basics

The technologies we collectively refer to as "blockchain" are a system for: (1) distributing work; and (2) coordinating that work across many computers connected by a network (usually the public Internet). This structure enables the coordination of distributed work without the need for a single centralized authority. Since this process heavily involves "verifying work" and need not necessarily use a "chain" style data structure,[14] we refer to these technologies collectively as "Distributed Verification Technologies" ("DVTs").[15] This Section gives a brief

the contract of sale and the rights and obligations of the seller and the buyer arising from such a contract. In particular, except as otherwise expressly provided in this Convention, it is not concerned with: (a) the validity of the contract or of any of its provisions or of any usage.").

[13] *See* Jarno Vanto, *Attorneys' Fees As Damages in International Commercial Litigation*, 15 PACE INT'L L. REV. 203, 218 (2003) ("[T]he preamble to the Convention provides: 'Being of the opinion that the adoption of uniform rules which govern contracts for the international sale of goods and taking into account the different social, economic and legal systems . . . .' This means that, at least on a symbolic level, the Convention takes into account different legal systems and consequently also the different outcomes they may produce." (citation omitted)).

[14] The data stored in a "blockchain" is an ordinal list of data points. *See* SATOSHI NAKAMOTO, BITCOIN: A PEER-TO-PEER ELECTRONIC CASH SYSTEM 2 (2008), https://bitcoin.org/bitcoin.pdf. More advanced data structures, such as variations on binary tree structures, can be expressed mathematically as combinations of linked lists. Accordingly, any data that can be expressed in a data structure of an abstract "list" type can therefore be expressed in a data structure of an abstract "tree" type. Similar analysis can be applied to other structures, such as matrices. *See generally* THOMAS H. CORMEN ET AL., INTRODUCTION TO ALGORITHMS, chs. 3, 5–6, 10 (3d ed. 2009).

[15] "Distributed Verification Technologies" is both more accurate and more usefully descriptive than the common business term "Distributed Ledger Technologies" for two reasons. First, the concept of a "ledger" implies a linked-list style data structure, which in addition to being only the prototype version of blockchain implementations, is also a comparatively highly inefficient data structure and thus is unlikely to be dominant (if even used) in the long-term. Second, the term "ledger" fails to capture the key element that

Electronic copy available at: https://ssrn.com/abstract=3533517

overview of "Blockchain Basics," and the following Section provides greater detail on the history and implementation of blockchain-based technology known as "smart contracts."

Such "peer-to-peer"[16] coordination is what enables the concept of "electronic cash" as bearer instruments—by arbitrarily distributing the *verification* of whether or not a particular cryptographic "token" properly belongs in a given "wallet." These "consensus" mechanism, blockchain-based cryptocurrencies allow those cryptographic tokens to become effective bearer instruments because no "owner" or "recipient" of a given token must depend on any specific or centralized party to confirm the validity of their "instrument" (token), but rather can look to the net product of the peer-to-peer system for such confirmation.

The peer-to-peer system, or consensus mechanism, can be implemented in a variety of different ways. Generally speaking, it comprises a data structure and an algorithm for community verification of modifications to the information stored in that data structure. In the context of most commonly used cryptocurrencies in 2020, the data structure usually comprises a singly linked list or public "ledger" of transaction history. Likewise, current blockchain implementations generally use a consensus algorithm that ensures a certain percentage of total "nodes" (computational participants[17]) in the network "agree" that a change should take place and ensures the accuracy of the ledger.[18] The threshold required for verification of an ownership transfer generally is sufficiently high that the cost of a "takeover"

facilitates distributed work—the verification process—and thus, we recommend that DVT is a superior term to describe this class of emerging technologies. *See generally* Usha R. Rodrigues, *Law and the Blockchain*, 104 IOWA L. REV. 679, 697 (2019) ("Blockchain technology, also called distributed ledger technology ('DLT'), offers four primary and related benefits: it is decentralized, it is transparent, it is (or at least can be) anonymous, and it is nearly impossible to manipulate."); GARRICK HILEMAN & MICHEL RAUCHS, CAMBRIDGE CTR. FOR ALT. FIN., GLOBAL BLOCKCHAIN BENCHMARKING STUDY 24 (2017), https://www.jbs.cam.ac.uk/faculty-research/centres/alternative-finance/publications/global-blockchain/#.X0f4EdNKhQI ("In general, the term 'distributed ledger technology' refers to all initiatives and projects that are building systems to enable the shared control over the evolution of data without a central party, with individual systems referred to as 'distributed ledgers.' If one wants to describe a system that has global data diffusion and/or uses a data structure of chained blocks, one should call it a 'blockchain.' However, 'blockchain technology' and 'distributed ledger technology' are still commonly used interchangeably despite attempts to semantically separate them by their different underlying architectures.").

[16] The term "peer-to-peer" is a slight misnomer in this context in the sense that most current implementations of blockchain technologies implement tiered peering systems, rather than true fully peer-to-peer systems. However, for the purposes of this Article's audience, the term is usefully descriptive.

[17] Often (but not always) individual "general purpose computers" or PCs, depend on the particular blockchain implementation. In certain blockchains, such as the well-known Bitcoin blockchain, the nature of the computations involved lend themselves to Application Specific Integrated Circuits ("ASICs"), and thus, the "nodes" in that regard may comprise highly specialized hardware instead of desktop or laptop PCs or servers running software as a background process.

[18] *See generally* Sarwar Sayeed & Hector Marco-Gisbert, *Assessing Blockchain Consensus and Security Mechanisms Against the 51% Attack*, APPLIED SCIS., Apr. 2019.

Electronic copy available at: https://ssrn.com/abstract=3533517

attack,[19] or inserting enough nodes to force an *unauthorized* transfer, will exceed the benefit of the value that can be obtained from the unauthorized transactions executed as a function of the "takeover." [20]

Thus, the key basic concept of blockchain technology is not necessarily so much about *currency* as it is about *verification*. This concept of distributed verification, of course, is a natural model for financial transactions like payments. This model has extensions far beyond mere currency exchanges[21] and, as this Article explores, may be a substitute for traditional common law contract dispute resolution.

B. Brief History of Blockchain Technology

The concept of blockchain was first proposed in a 2008 paper titled *Bitcoin: A Peer-to-Peer Electronic Cash System* and published under the pseudonym Satoshi Nakamoto.[22] The paper proposed a solution for authenticating digital transactions without the need for a centralized authority. It articulated a series of proofs for a cash-like electronic commerce system that could be maintained by an arbitrarily distributed network with no centralized verification.[23] These proofs showed that a distributed verification algorithm, or "consensus" mechanism, could prevent the double spending of digital assets. The "double spend problem" occurs when digital tokens—which can be copied flawlessly—are used to represent value. Because such tokens can be copied flawlessly, any given token, i.e., any digital monetary instrument, could be used twice or more, and the payee would have no mechanism of determining which was the original. The consensus mechanism distributes the verification of payments across many different computers, forming a so-called "digital ledger" that verifies the authenticity and provenance of payments before

[19] This bears some conceptual similarity to a hostile takeover in the sense that at least some majority control will usually be required to execute the attack. However, given the fact that only a currency acquisition (as opposed to productive assets of a company) can be acquired, such an attack would at best only be a speculative investment in the cryptocurrency context and one that is extremely unlikely to be profitable since the market almost certainly would rapidly *devalue* a cryptocurrency compromised in that fashion.

[20] *See* JOSHUA A. KROLL ET AL., THE ECONOMICS OF BITCOIN MINING, OR BITCOIN IN THE PRESENCE OF ADVERSARIES 12 (2013), https://www.econinfosec.org/archive/weis2013/papers/KrollDaveyFeltenWEIS2013.pdf ("As a cartel must outmine the entire Bitcoin network and thus outspend the entire Bitcoin network for as long as it would remain a cartel, we believe it is very unlikely that a cartel could double-spend enough to recover the cost of the attack."). *But cf.* Sayeed & Marco-Gisbert, *supra* note 18, at 9 (assessing this probability differently and stating that "majority hash attacks have been a serious problem in recent times"). The Authors disagree with that assessment in that Sayeed and Marco-Gisbert's analysis is context-dependent on specific market conditions, rather than inherent to the scientific design, and thus seems unlikely to apply to the general case. Kroll et al.'s analysis is more consistent with the Authors' analysis. *See* KROLL ET AL., *supra*.

[21] *See generally* DAVID THAW & WILL KANG, OBNOSTIC: AN OBJECT-AGNOSTIC "GENERAL PURPOSE BLOCKCHAIN" (2019), https://47b16f07-4bfb-43f2-9be9-5c47db516f53.filesusr.com/ugd/b86d62_2da0ae7bb43d40a08ca0e7b2926a1647.pdf; U.S. Provisional Utility Patent Application No. 62/792,381 (filed Jan. 15, 2019).

[22] NAKAMOTO, *supra* note 14.

[23] *Id.*

Electronic copy available at: https://ssrn.com/abstract=3533517

settlement, and thereby prevents "double spending."[24] Nakamoto's paper called the hypothetical system Bitcoin. The Bitcoin system prototype was deployed worldwide soon after the paper's publication[25] and gave rise to the name of the asset currently traded worldwide under the symbol "BTC."[26] The Bitcoin system prototype was deployed worldwide soon thereafter.

Nakamoto's paper and subsequent implementation of the first Bitcoin prototype in 2009 became the foundation of the blockchain concept. Together the paper and implementation proved the viability of DVTs (more commonly referred to as "Distributed Ledger Technologies").[27] More concisely, DVTs are distributed networks for maintaining consensus about data and making decisions regarding operations. They distribute computational work to maintain agreement about the answers to a set of questions without the need for intervention or verification by a centralized authority.

Some background and history on the concept of blockchain (or DVTs) will be instructive to help contextualize the examples in which we apply common law contract doctrine to current implementations of smart contracts.

1. The DVT Concept and the Development of Blockchain

Much of blockchain's success is attributed to its distributed nature, specifically the ability to maintain "trust" in the origin and ownership of digital objects without the need to rely on a centralized authority.[28] This concept of distributed verification, or "consensus," is the core of the blockchain concept. It comprises two essential elements: (1) a distributed data structure to maintain information;[29] and (2) a consensus mechanism (algorithm) to govern and prove the authenticity of work performed by nodes participating in the distributed network.[30] The distributed data structure allows the DVT to work across many computers around the world without needing any one single computer or "central authority" to

[24] *Id.*

[25] Eric D. Chason, *How Bitcoin Functions as Property Law*, 49 SETON HALL L. REV. 129, 131 (2018) ("Barely two months [after Nakamoto's paper was published], in early January 2009, Bitcoin 'went live' with the creation of the first units of Bitcoin.").

[26] *Id.* at 132–33.

[27] "Distributed Ledger Technologies" ("DLTs") is, by far, a more commonly accepted term for the group of technologies used in cryptocurrencies and smart contract platforms. Although not the primary subject of this Article, we note that this term is technologically limiting in that it only contemplates globally maintained ledgers, as opposed to the larger category of efforts enabled by this technology—the ability to make coordinated decisions without the need for a firm or other centralized authority. In this context, "Distributed Verification Technologies" is a much more usefully descriptive term, and also better identifies why future technologies may have more ability to implement the guarantees of common law contracts. *See infra* Sections II.A–B.

[28] *See generally* SHAWN S. AMUIAL ET AL., THE BLOCKCHAIN: A GUIDE FOR LEGAL AND BUSINESS PROFESSIONALS § 1:2 (2016).

[29] *See* NAKAMOTO, *supra* note 14, at 3.

[30] *See id.* at 4, 8.

Electronic copy available at: https://ssrn.com/abstract=3533517

coordinate the participants.[31] The system enables the participants to maintain "consensus," a term which represents the algorithmic process by which the computers participating in a given blockchain reach "agreement" regarding a specific data point, e.g., whether or not a transaction occurred or who owns a particular portion of a given Bitcoin.[32]

A DVT is capable of maintaining consensus regarding data without an agreed upon common authority. DVTs thus can enable collective "answering of questions" and maintenance of data records. DVTs also provide the data structure which stores (or links to) the data they authenticate.[33] The combination of these two items together—a distributed verification network and a distributed data structure—forms the DVT itself.

a. "Blockchain 1.0": Static Digital Tokens

The Bitcoin blockchain was the first example of a DVT. It encapsulates the idea of the "Blockchain 1.0" concept—a distributed verification of static tokens. The Bitcoin tokens ("BTCs") are, quite literally, static cryptographic tokens which can be exchanged among "owners" via a provably hard[34] public key cryptographic system. The public–private keypair cryptographic system prevents unauthorized transfers, and the Bitcoin network (the DVT) maintains a "ledger" of all Bitcoin transactions since the network's inception, preventing duplication of the tokens.[35]

[31] The coordination function, rather, is distributed across the entire network. Interestingly, this structure parallels work on the Theory of the Firm in economics, and some scholars have suggested that a similar concept can be used to create decentralized organizations, sometimes referred to as Distributed Autonomous Organizations ("DAOs"). Kevin Werbach & Nicolas Cornell, *Contracts Ex Machina*, 67 DUKE L.J. 313, 337 (2017) ("If a corporation is simply a nexus of contracts, why not encode those agreements into digital self-enforcing agreements? A DAO could have stock ownership, corporate governance rules, payroll arrangements, and virtually all of the economic trappings of a modern corporation, all running automatically in a completely distributed manner.").

[32] *See* NAKAMOTO, *supra* note 14, at 8.

[33] *See id*. at 5.

[34] "Provably hard" is a term of art in computer science and related fields which refers to the computational complexity, or "difficulty," of solving a certain type of problem. It generally refers to the time it takes to solve such a problem through "brute force" guessing, expressed as a function of the number of guesses required to find a working solution. Since, technically speaking, a brute force mechanism has equal probability of being successful on the first try (a trivially short period of time) or on the last possible try (an extremely unlikely result), mathematical equations are instead used to express the "average case." This concept of an "average case," i.e., average time to guess, focuses not on a specific amount, but rather the degree of difficulty of the equation which models the "average" number of guesses. The result divides difficulty into categories, such as "constant time," "linear time," "polynomial time," "exponential time," "factorial time," and so on. With classical (non-quantum) computers, problems that require more than polynomial time to brute force generally are considered to be "hard" because the combined computational power of all computers on earth is insufficient to solve such problems in reasonable time (less than millennia) by brute force guessing. This is a critical distinction because—much like the function of modern encryption—it is the mechanism by which computers "prevent" unauthorized changes or access in systems which must share data publicly, such as cryptocurrencies and other blockchain-style DVTs. *See generally* CORMEN ET AL., *supra* note 14.

[35] *See* NAKAMOTO, *supra* note 14, at 2.

Electronic copy available at: https://ssrn.com/abstract=3533517

The result is a first-generation form of digital private property—static tokens which can be "owned" and "possessed" like tangible property in the real world.

These concepts of ownership and possession, without the need for mediation or approval by a centralized authority, have been cited by many observers as key drivers of the popularity of cryptocurrencies.[36] Both privacy concerns and cybersecurity concerns are among those driving cryptocurrency popularity.[37] Among individuals who distrust centralized currency and payment systems based on fiat currencies—whether for legitimate[38] or illegitimate[39] reasons—the decentralized, distributed nature of blockchain-based cryptocurrencies is very attractive. From a security standpoint, individuals concerned with the vulnerabilities of centralized single-point-of-failure systems for electronic payments[40] or with the stabilities of fiat currencies[41] may find the decentralized aspects of cryptocurrencies

[36] *See, e.g.*, Jorge Galavis, *Blame It on the Blockchain: Cryptocurrencies Boom Amidst Global Regulations*, 26 U. MIAMI INT'L & COMP. L. REV. 561, 564 (2019) ("[T]he underregulated market is likely the reason for the booming popularity of ICOs and Cryptocurrencies in general."); Alice Huang, *Reaching Within Silk Road: The Need for a New Subpoena Power that Targets Illegal Bitcoin Transactions*, 56 B.C. L. REV. 2093, 2101–02 (2015) ("Due to Bitcoin's growing popularity and its advantages over traditional payment methods, many businesses have begun accepting the virtual currency. One crucial advantage is the payment freedom that Bitcoin provides. Transactions are instantaneous and borderless; unlike banks, which restrict users by business hours, holidays, and transfer limits, Bitcoin does not impose any limitations on the time, place, or amount of its transactions. Furthermore, Bitcoin has very low transaction fees and sellers have the ability to bypass the usual cost of accepting a credit card payment." (citations omitted)).

[37] *See* Huang, *supra* note 36, at 2103 ("One of the main reasons Bitcoin has become popular is the near anonymity it offers. Users are virtually anonymous because its public key encryptions only reference the locations of bitcoins without disclosing any other information about the user. In this sense, Bitcoin is analogous to cash. Each transaction is neatly recorded but it becomes difficult for government officials to identify the individuals behind the transactions." (citations omitted)).

[38] *See, e.g.*, Jonathan B. Turpin, *Bitcoin: The Economic Case for a Global, Virtual Currency Operating in an Unexplored Legal Framework*, 21 IND. J. GLOBAL LEGAL STUD. 335, 360 (2014) ("Additionally, dissidents in oppressive countries may find Bitcoin to be a preferred method of payment for their opposition activities. For example, an anti-government blogger in China must take great care to avoid being identified by the highly skilled Internet police.").

[39] *See, e.g.*, Carmine DiPiero, *Deciphering Cryptocurrency: Shining a Light on the Deep Dark Web*, 2017 U. ILL. L. REV. 1267, 1269 (2017) ("'Silk Road' was one of the first 'Darknet markets' to emerge on the Internet since the invention of the bitcoin. Throughout 2011 and 2013, users of the Silk Road website could buy anything—drugs, child pornography, arranged murders, hacked credit cards, and countless other illicit activities and substances—using a virtual currency known as 'bitcoin.'").

[40] *See, e.g.*, Emily Flitter & Karen Weise, *Capital One Data Breach Compromises Data of Over 100 Million*, N.Y. TIMES (Jul. 29, 2019), https://www.nytimes.com/2019/07/29/business/capital-one-data-breach-hacked.html.

[41] *See, e.g.*, Swati Goyal, *The Difference Between Fiat Money and Cryptocurrencies*, YAHOO FIN. (Aug. 9, 2018), https://finance.yahoo.com/news/difference-between-fiat-money-cryptocurrencies-132027811.html ("[Fiat] currencies are always at risk of becoming worthless due to hyperinflation as they are not linked to any physical reserves such as commodities.").

Electronic copy available at: https://ssrn.com/abstract=3533517

equally attractive. In both cases, cryptocurrencies are acting like tangible property and performing an economic function largely suppressed by the digitization of society—the ability to engage in payment-based transactions without having to reveal one's identity or create a permanent record of the transaction associated with an individual.[42] Historically, anonymity was accomplished through "cash" transactions (or "cash equivalents" like diamonds, in the case of those who distrust central currencies) because of the bearer-instrument-like nature of such currencies or commodities. The absence of bearer-instrument-like property rights in the digital economy had previously prevented individuals from transacting anonymously via Internet-based commerce. The (re)introduction of these rights through cryptocurrencies restored such transaction capacity.

DVTs thus create a class of potential rights in digital objects similar to that applicable to tangible property under classic common law property doctrine.[43] Because the information contained within the tokens need not *necessarily* represent a currency object, the potential for enabling traditional property rights in other types of digital property[44] follows logically from this analysis. We note these in the following portions of this Section addressing advances in DVTs and plan to address this concept further in future work.

Ironically, Nakamoto did not reference the term blockchain anywhere in the paper and only described the concept of a "proof-of-work" algorithm.[45] Lack of information regarding Nakamoto's true identity currently precludes determining whether Nakamoto recognized the potential of Bitcoin as a broader concept or for that matter, whether Nakamoto has participated in any additional blockchain or DVT projects.

[42] Whether identified or not, even "anonymous" online transactions may still create permanent records associated with a *unique* individual, even if that individual is not associated with a specific natural person. As privacy scholars have observed, (re)identification or deanonymization of such activities may subsequently lead to the association of those previously anonymous individual online personas with physical-world natural persons. *See, e.g.*, Steven M. Bellovin et al., *Privacy and Synthetic Datasets*, 22 STAN. TECH. L. REV. 1, 13, 16 (2019) ("In actuality, identifying individuals using seemingly non-unique identifiers is far easier than a data sanitizer might hope. . . . To be sure, it is difficult to pin down exactly what data identifies individuals, but it is even more difficult to accurately predict what potential auxiliary information could be available in the future—i.e., the de-identification-re-identification arms race.").

[43] *See, e.g.*, Doug Fredrick, *Down the Rabbit Hole: Cryptocurrency & Blockchain*, WIS. LAW., Mar. 2019, at 22, 28–29 ("The sequential nature of blockchain networks naturally lends itself to transactions such as recording documents that transfer ownership of real estate or vehicles, as well as maintaining medical records. Recorder or register of deeds' offices are especially predisposed to implementation of blockchain technology, because a large part of what they do is maintain a public ledger, so it would be a relatively small step to digitize and automate the process of recording real estate documents and even marriage certificates.").

[44] E.g., electronic records (such as health or financial records), digital creative content, etc.

[45] In fact, the word "consensus" appears only once in the paper's conclusion. *See* NAKAMOTO, *supra* note 14.

Electronic copy available at: https://ssrn.com/abstract=3533517

Although the computational engine which operates the Bitcoin network does have (at least some) ability to perform arbitrary computation, to the extent that ability exists it is constrained as a matter of implementation.[46] Practically speaking, therefore, Bitcoin and other "Blockchain 1.0" implementations generally can do little more than provide the tangible property rights described above for fixed, predefined (static) tokens.

b. "Blockchain 2.0": Executable Agreements ("Smart Contracts")

The next step in the evolution of DVTs was the ability to implement arbitrary computation, or the development of what computer science refers to as a "Turing-complete" virtual machine ("VM").[47] Such a VM was unnecessary for the limited-purpose DVT implementations of most static "Blockchain 1.0" tokens known as cryptocurrency.[48] Because fixed-token cryptocurrency implementations had predetermined and (generally) permanently fixed computational models,[49] the arbitrary applications supported by a full VM were unnecessary.[50]

As these cryptocurrencies gained popularity in online commerce, however, the market increasingly sought methods to computationally enforce *agreements* surrounding the exchange of those tokens similar to how "Blockchain 1.0" cryptocurrencies computationally enforced *ownership* of tokens. A growing desire to automate certain cryptocurrency transactions based on various conditions being satisfied drove a need for more computational flexibility to implement apps which could accomplish this automation. For example, two parties might want to schedule a payment to occur on a monthly basis in exchange for a product or service, where

[46] *See* CRAIG S. WRIGHT, BITCOIN: A TOTAL TURING MACHINE 243 (2017), https://ssrn.com/abstract=3265146.

[47] *See generally id.* Arbitrary computation is the ability to perform any type of computational operation on a particular system—usually through an operating system which provides an interface to translate commands from a programming language into physical-level instructions. Nonarbitrary computers, or ASICs are by contrast capable only of performing one specific type of operation. ASICs are popular as Bitcoin "miners" because they can be constructed to perform SHA-256 operations (the mining algorithm) much faster if the processor is built *only* to perform such calculations. A less restrictive but similar concept is found in Graphics Processing Units ("GPUs") of many modern computer systems which have a dedicated "chip" (microprocessor) separate from the main processor (CPU) which is designed for and dedicated to graphics operations.

[48] *See* AMUIAL ET AL., *supra* note 28, § 2:3 ("[S]mart contracts represent a significant advance over the basic scripting language that only maintains unspent transaction outputs on a distributed ledger (e.g., Bitcoin). While the Bitcoin protocol contains a basic scripting language that allows for some programming functionality, it is not nearly as robust as the Ethereum Virtual Machine (EVM) that is incorporated into the Ethereum Protocol or similar protocols with Turing-complete programming capabilities.").

[49] *See generally supra* note 47 for a discussion of the concepts of arbitrary (general purpose) computing versus application-specific computing. "Fixed" computational models are those which can be designed for the latter category through the development of ASICs.

[50] This is similar to the concept of the computation chip in a hand calculator versus that found in a laptop computer. The former performs a specific, predefined set of computations, whereas the latter is a "general purpose computer" that may need to run any number of arbitrary "apps" for the user.

Electronic copy available at: https://ssrn.com/abstract=3533517

the governing app would automatically transfer a specified amount of cryptocurrency from the buyer to the seller once the product or service was delivered. Essentially, the market wanted DVTs to implement self-executing agreements or contracts.[51]

This demand led to the development of platforms like Ethereum to implement what became known as "smart contracts." Ethereum is a DVT that implements a Turing-complete VM, the Ethereum Virtual Machine ("EVM"), capable of arbitrary computation. Ethereum allows users to build apps on a platform which combines token ownership with encodable agreements describing when and how ownership will change. Implementing this type of encodable agreement (e.g., smart contract) requires a programmatic language specifying terms and conditions under which a token-based transaction will take place. Thus, implementing smart contracts effectively requires arbitrary computation to allow a contracting language with sufficient flexibility. The "fixed computational" approach of Bitcoin or similar systems is insufficient for this goal.

Ethereum represented a substantial advance forward in the development of DVTs. Parties to cryptocurrency transactions could encode the terms of those transactions in Solidity[52] and have those terms become part of the Ethereum blockchain. The details of these terms would then be verifiable by the Ethereum consensus mechanism similar to how ownership of tokens is verifiable by "Blockchain 1.0" consensus mechanisms. These smart contracts could ensure that once specified computationally verifiable circumstances are satisfied, the terms of the agreement would be fulfilled and the associated cryptocurrency (i.e., ETH tokens) would be transferred. The effective result is a self-executing agreement.

Such smart contracts, however, are generally immutable once accepted into the ETH blockchain and, as such, may not be able to grant the full spectrum of legal contract rights. Most notably, they are not subject to post-execution redress mechanisms which appear in law but not within the contract themselves. Put simply, the computational nature of smart contracts quite literally constrains any resolution of the agreement terms to the "four corners of the contract" (code) itself.

Because the computational function of this verification is managed by the EVM, it is technically possible to create arbitrary programs which run on the

[51] *See* Werbach & Cornell, *supra* note 31, at 317 ("Firms can achieve significant cost savings and efficiency gains when using computers to automate contracting.") (citing JAMES SCHNEIDER ET AL., GOLDMAN SACHS, PROFILES IN INNOVATION: BLOCKCHAIN (2016), https://pgcoin.tech/wp-content/uploads/2018/06/blockchain-paper.pdf (detailing different applications and cost-saving estimates of blockchain across several industries)).

[52] *See generally* SOLIDITY, https://solidity.readthedocs.io/en/v0.7.0/ (last visited Aug. 31, 2020) ("Solidity is an object-oriented, high-level language for implementing smart contracts. Smart contracts are programs which govern the behaviour of accounts within the Ethereum state. Solidity was influenced by C++, Python and JavaScript and is designed to target the Ethereum Virtual Machine (EVM). Solidity is statically typed, supports inheritance, libraries and complex user-defined types among other features. With Solidity you can create contracts for uses such as voting, crowdfunding, blind auctions, and multi-signature wallets.").

Electronic copy available at: https://ssrn.com/abstract=3533517

Ethereum network.[53] This functionality was one of the goals of Ethereum's founder, Vitalik Buterin, who initiated the project as an offshoot of Bitcoin after years of unsuccessfully arguing that Bitcoin needed a more general scripting language as part of its DVT platform.[54] While Ethereum does technically implement this capability, it is limited by two critical design constraints. First, the EVM does not scale—it lacks the computational capacity to handle more complex distributed applications ("dApps").[55] As some blockchain developers have joked, "Ethereum runs scared at the sight of a baby kitten."[56] Second, the EVM is designed to implement contracts, and as such, the language is constructed with that purpose in mind. The result is reminiscent of Roman contract law—"you can build any dApp you want on Ethereum, as long as it is a dApp comprising Smart Contracts."[57]

An additional shortcoming of Ethereum—and other "Blockchain 2.0"—based smart contracts is the need to rely on a concept known as "Oracles." The limitations of the EVM restrict the information contained "on-chain," or as part of the blockchain data structure itself, to that information which is encoded into the executable Solidity code and the ETH tokens themselves.[58] In other words, a smart contract only "knows" the information contained within it when it is "written." But what if the contract execution depends on some external information, such as completion of a project (as most contracts do)? In these cases, smart contracts must rely on a concept known as "Oracles"—external data sources programmed into the smart contract itself. These data sources can be virtually anything accessible from

[53] *See* Werbach & Cornell, *supra* note 31, at 333–34 ("Ethereum's scripting language is significantly more powerful than Bitcoin's. It is Turing complete, which means it can in theory execute any function that can be processed by a computer.").

[54] *See* VITALIK BUTERIN, ETHEREUM, A NEXT GENERATION SMART CONTRACT & DECENTRALIZED APPLICATION PLATFORM (2015), https://pdfs.semanticscholar.org/0dbb/8a54ca5066b82fa086bbf5db4c54b947719a.pdf?_ga=2.104118087.655109952.1597800835-1475162564.1597800835.

[55] Note that some within the community refer to "dApps" as decentralized applications. Regardless of which terminology one selects, the lack of scalability of the EVM remains.

[56] This is a reference to the Cryptokitties app, a blockchain-based virtual game for the "growing," trading, and "raising" of virtual cats. In December 2017, the dApp nearly brought the Ethereum network to a halt by overpowering the computational capacity of the network since the dApp's complexity exceeded the computational power traditionally envisioned for smart contract evaluation and settlement. *See CryptoKitties Craze Slows Down Transactions on Ethereum*, BBC NEWS (Dec. 5, 2017), https://www.bbc.com/news/technology-42237162.

[57] Several scientists, business leaders, and legal scholars have noted that physical-world organizations are, in fact, just a collection of contractual relationships. Putting aside the elements of psychology and sociology dealing with the inherently non-deterministic nature of human behavior, this view—one often advocated by proponents of DAOs—ignores the inherently *qualitative* aspects of human decision-making, contractual evaluation, and organizational operation for any organization which comprises *humans as well as contracts* (which is, by definition, all physical-world organizations).

[58] It is, of course, possible to create programs within the EVM that store additional data, but for the same reasons EVM has difficulty creating dApp—primarily scalability issues—EVM has difficulty storing large amounts of data.

Electronic copy available at: https://ssrn.com/abstract=3533517

the public Internet, but unlike other parts of the smart contract, the EVM *cannot inherently verify the authenticity of any information provided by an Oracle*.[59]

The failure of "Blockchain 2.0" projects, most notably Ethereum, to realize the full potential of DVTs has sparked offshoot projects in an attempt to correct these two key limitations and implement additional features. Most notably, the EOS.IO project ("EOS")—a multi-billion (U.S.) dollar organization—is attempting to build a true distributed computational platform with all the traditional aspects of a computer.[60] If successful, this "Blockchain 2.5" project would represent a substantial step towards fully realizing the potential of DVTs. Such projects also represent opportunities to develop implementations of the other examples of common law property rights in digital objects alluded to earlier in this Section. EOS, for example, implements trading of computational resources such as general processing,[61] graphics processing,[62] and volatile memory.[63]

C. Present Smart Contract Limitations

Current implementations of smart contracts can accomplish many things which create meaningful economic value. They can reduce transaction costs for parties by permitting individuals to create "self-settling" agreements without the need for further performance solely for contract settlement purposes. They can reduce uncertainty by creating programmatic options which specify when and how settlement *will* occur without the need for third-party escrow.[64] These and similar

[59] *See generally* John R. Kosinski, *Ethereum Oracle Contracts: Setup and Orientation*, TOPTAL: DEVELOPERS, https://www.toptal.com/ethereum/ethereum-oracle-contracts-tutorial-pt1 (last visited Aug. 27, 2020). Note also, as does Kosinski, that "[t]he smart contract space, being so new, changes quickly . . . [and] features that were new when this article was written may be deprecated or obsolete by the time you're reading this." *Id.*

[60] *See generally* EOSIO, https://eos.io/why-eosio/ (last visited Sept. 1, 2020).

[61] General purpose computational processing is that performed by integrated circuits colloquially known as CPUs, which are not considered to be application-specific and can perform arbitrary computation, but generally with less efficiency for a given application than would a "chip" designed specifically for a given application.

[62] Graphics processing for the display of multimedia on various visual devices, e.g., screens, VR headsets, holography, etc., are an example of a more application-specific integrated circuits, known colloquially as GPUs, where the physical hardware is designed to perform a more limited set of computations—generally with much greater efficiency—than a CPU.

[63] Volatile memory is a term-of-art referring to the "temporary" memory storage of a computing device. Historically the colloquial term for this was a computer's random access memory ("RAM"), as distinguishable from the floppy disks, optical discs, or hard disks or a computer. The latter—collectively described as "non-volatile" forms of memory—were used for "long term" storage after a computer was powered down or reset. In modern practice, these distinctions have blurred as almost all forms of long-term storage now have the full random access capabilities of short-term volatile memory, and the physical hardware for both are similar, i.e., both use solid-state "chips" rather than magnetic or optical "discs."

[64] James Grimmelman, *All Smart Contracts Are Ambiguous*, 2 J.L. & INNOVATION 1, 20 (2019) (noting that "smart contracts" do not completely eliminate uncertainty because potential semantic changes can be both unexpected and devastating and that "[b]lockchain-based smart-contract programming languages don't have continual linguistic drift; they have occasional earthquakes").

Electronic copy available at: https://ssrn.com/abstract=3533517

features of "self-executing" agreements are all laudable, beneficial technological advances. We do not argue that these advances are immaterial or that they may not lead to further development.[65] But the implications for parties using technologies in their current form must be evaluated in terms of the expectations of those parties. And the term "smart contract" may create an expectation or understanding of what a contract is and what it means to be smart. Current technology used to run smart contracts cannot meet those expectations. The following Part explains this disconnect.

III. CONSEQUENCES OF "SMART" CONTRACTING ARE BETTER UNDERSTOOD AS "BOT" CONTRACTING

The limitations of current smart contract technology ("Blockchain 2.0") have important implications for the application of common law contract doctrine to this relatively new type of automated agreement. The fixed nature of these transactions leads to two consequences that are belied by the "smart contracts" moniker. First, smart contracts are not actually contracts in the common law sense of the term because they eliminate the role of courts or other neutral arbiters to resolve any disputes that may arise. The nature of the technology literally prevents the parties' ability to breach the contract. Therefore, current smart contract technology is not "smart" because unlike a traditional contract, it is inflexible and cannot account for subject matter and capacity requirements of common law contract doctrine. Statutory law and common law contracts doctrine have evolved to articulate numerous reasons why a contract may be void or voidable after execution, most of which cannot be accounted for by current smart contract technology. However, the automated features that make these transactions different from typical contracts are what give these transactions their biggest benefits.

A. The Elimination of an Enforcement Authority

Just as a doctrinal determination regarding the status of smart contracts is premature, we argue that the usage of this term also is premature. Our contention is that the moniker "smart contracts" may be taken to believe that courts can intervene in circumstances where, in fact, a court's ability to award damages or enforce the agreement is not technologically feasible. The word "contract" may lead consumers to believe that common law contract defenses are available when in fact they are not. A better name for these transactions would be "bot contracts," as "bot" connotes the fixed, Internet-based nature of these deals. Global usage of the term "smart contract" combined with the rate of technological change is likely to make adoption of a new generic category unsuccessful. Nonetheless, legal scholars, practitioners, and courts should understand the benefits and limitations of these technological tools to avoid the trap of assuming the term "smart contract" connotes capabilities these devices currently cannot deliver.[66]

[65] Quite the contrary, as discussed in Part III, it seems more likely that current technologies will lead to future developments capable of implementing many, if not all, of the elements of common law contract doctrine.

[66] This point is particularly salient given that smart contracts have emerged in commercial use most predominantly in civil law jurisdictions such as the Republic of Korea,

Electronic copy available at: https://ssrn.com/abstract=3533517

It is worth noting that the ability to enter into such automated contracts is fully defensible and consistent with freedom in contracting. For example, one might argue that a smart contract really is a choice by the parties to have a self-settling agreement that becomes instantly binding, similar to the acceptance of an offer on the floor of a stock exchange or the motion signaling a bid at an auction. While it is true that in both cases such actions do signal the formation of a *presumptively* self-settling contract, in both contexts the settlement of that contract *can* be interrupted for proper causes such as fraud or duress. Thus, it is possible that the term "smart contract" would lead consumers to believe that these so-called "smart" agreements allow for similar interruption capabilities. However, this confusion would be avoided with the use of a generic term that more accurately reflects the limitations of these agreements that defy normal contracting expectations.

For example, if a smart contract is used to implement an auction and the goods are discovered to be fraudulent before settlement, no authority will have the *ability* to intervene and prevent the contract from executing unless that ability is programmed into the executable code of the smart contract *before* it is committed to the blockchain. This is inconsistent with the law surrounding fraud, which enables courts to free litigants from obligations procured fraudulently before full performance has been completed.[67]

One might also claim that the failure of a given smart contract to implement "safeguards" which allow for judicial intervention is a drafting defect that implies the parties have assumed the liability for such defects by electing to use this technology. Stated differently, the response might be that the parties have impliedly waived these defenses by using this transaction method, and if they do not want to do so in the future, they may clarify their desires by writing better code, just as courts often admonish litigants to memorialize their intent in clearly written contracts.[68] There is some merit to this argument, to the extent that (at least in common law jurisdictions) contract drafting has developed extensively, and smart contract coding is in its infancy. Indeed, contract drafting has developed so far that some have argued there is insufficient time for the average person to keep up with all the contracts to

Japan, and Germany. While common law jurisdictions (most notably the United Kingdom) have expressed some interest in these areas, the United States and United Kingdom are remarkably behind the technological curve in the development and utilization of blockchain technologies. Given the contract doctrine differences present between common law and civil code jurisdictions, it is worth noting that the moniker "smart contracts" may have gained traction more easily in those nations because current technologies may be able to implement more of the doctrinal requirements of contracts under local law in civil code jurisdictions than in common law jurisdictions.

[67] For the purposes of this argument, we assume fraud defenses to be nonwaivable. We recognize that this is not necessarily the case in all circumstances, but the prevalence of nonwaivable fraud protections in a majority of common law and civil code jurisdictions makes those edge cases relatively immaterial to this argument.

[68] Effects Assocs., Inc. v. Cohen, 908 F.2d 555, 557 (9th Cir. 1990) ("Common sense tells us that agreements should routinely be put in writing. This simple practice prevents misunderstandings by spelling out the terms of a deal in black and white, forces parties to clarify their thinking and consider problems that could potentially arise, and encourages them to take their promises seriously because it's harder to backtrack on a written contract than on an oral one.").

Electronic copy available at: https://ssrn.com/abstract=3533517

which they are subject.[69] The use of smart contracts with standardized code (which could easily be evaluated for its "standard terms" at low cost to a user) could represent substantial transaction cost reduction in this context, particularly in industries where standard commercial terms are common to many or most transactions.

The "write better code" argument has its limits, however. Standard default rules can add efficiency. Statutory and common law contain many default provisions that can be imported by courts as gap-fillers and therefore eliminate the necessity of spelling out every conceivable term. As discussed above, one need not claim the right of fraud protection in advance to enjoy such protection, and indeed in most cases this protection is nonwaivable. A similar analysis would apply to other void-for-public-policy defenses and to duress defenses. Furthermore, code is deterministic,[70] and current technological limitations preclude intervention by the courts based on qualitative examination of an unpredictable condition. Therefore, the "write better code" argument fails in circumstances where the transaction has nondeterministic aspects to it, which may occur because the parties lack the ability under current technology to account for those aspects in the coding of their smart contract.[71] The only solution would be to have a general "interrupt" in which either party could, upon the proper initiation of legal action, cause the smart contract not to resolve. However, such an approach is effectively no different than current written contracts and carries similarly high transaction costs. Thus, while this approach might mitigate some doctrinal concerns, it would do so at the cost of much of the practical benefits associated with bot contracts.

Another common counterargument might be that the court's ability to order money damages is not reduced. If something invalidates a contract before settlement, and the contract is self-settling, the court simply would order the transaction reversed, or if not practically feasible, it would order appropriate money damages in the form of a second transaction. This response is informative and recognizes important freedom of contract principles but fails to recognize fully the limits of existing smart contract technologies. The design of existing blockchains is such that generally speaking, it is impossible to physically compel payment in the absence of cooperation by a breaching party. If, for example, a breaching party is judgment proof (far from a wild hypothetical in the current cryptocurrency environment) and elects not to cooperate, the private-key cryptographic aspects of the blockchain will prohibit any effective compulsory financial enforcement. The options in this scenario may be vastly different from the array of remedies available to litigants who seek to enforce financial promises through judgments or arbitration awards based on fiat currencies. Traditional litigants often implicitly rely on a

[69] *See* Aleecia M. McDonald & Lorrie Faith Cranor, *The Cost of Reading Privacy Policies*, 4 I/S: J.L. & POL'Y FOR INFO. SOC'Y 543, 565 (2009) ("We estimate that reading privacy policies carries costs in time of approximately 201 hours a year, worth about $3,534 annually per American Internet user. Nationally, if Americans were to read online privacy policies word-for-word, we estimate the value of time lost as about $781 billion annually.").

[70] Loosely defined as "capable of being mathematically predicted in advance," or alternatively stated, not subject to randomness.

[71] *See* Edmund Schuster, *Cloud Crypto Land* 15–16 (London Sch. of Econ. Legal Stud., Working Paper No. 17/2019, 2018), https://ssrn.com/abstract=3476678.

Electronic copy available at: https://ssrn.com/abstract=3533517

financial transaction ecosystem where financial institutions, employers, and other participants are bound by regulatory and criminal law to enforce judicial orders and freeze or otherwise dispose of funds in satisfaction of a judgment collection order. Courts currently have no such ability to effect orders of this type in the smart contract context.[72]

Finally, we note that the deterministic nature of current smart contract executable code may be generally unsuitable for agreements which involve substantial qualitative judgments, such as transactions in artistic objects, professional services, or architectural design. In such circumstances, not only would current smart contract technology be required to rely substantially on Oracles,[73] but also the Oracle's judgment would effectively become that of a binding arbiter. This is not to argue that the parties could not agree to this,[74] but rather that the parties must understand that the role of the Oracle is transforming from the mere "informative" one most parties would likely expect (similar to the role of an appraiser in a real estate transaction) into the role of a binding arbiter which would largely deprive the parties of their rights to seek judicial redress.

Many of these arguments and implications for smart contract users center around the disconnect between the current market *impression* of those technologies and the general market expectations of individual parties (at least in common law jurisdictions). We fully acknowledge that these may change. Our argument regards the current state of the market and that the term "Bot Contracts" more accurately describes the current state of technology and the types of expectations market participants should anticipate when using "Blockchain 2.0" agreements based on platforms such as Ethereum, EOS, NEO, and related projects.

B. Smart Contracts Neutralize Legal Defenses

Not only are smart contracts not smart, they also are wholly amoral. They will execute no matter what criminal, fraudulent, or extortionist acts may underlie their foundation. The enforceability of common law contracts depends on the notion that the agreement concerns subject matter that courts deem worthy of enforcing and that the actors have the capacity to perform. Without these two features, a contract

[72] Of course, as discussed in Section III.B, future developments in technology could implement these features. Indeed, the Venezuelan government is attempting to do so. *See, e.g.*, Darren Parkin, *Venezuela's President Orders Banks to Open Crypto Desks*, EXPRESS (July 5, 2019), https://www.express.co.uk/finance/city/1149744/crypto-desks-banks-venezuela-nicolas-maduro-crv. However, for parties currently considering using smart contract technologies, and for dApps developers considering building platforms based on existing technologies, these considerations should weigh heavily into determining the types of transactions they elect to support or use smart contracts for.

[73] In the interest of full disclosure, the Authors note that David Thaw currently is involved with a research and development project oriented toward implementing an "on-chain" solution for automating certain classes of qualitative evaluations of this nature.

[74] Indeed, many courts have upheld binding arbitration clauses, even in the context of class actions (or waivers thereof). *But see, e.g.*, Lamps Plus, Inc. v. Varela, 139 S. Ct. 1407 (2019) (affirming district court order that compelled arbitration but dismissing class-action arbitration claims against employer from approximately 1,300 employees whose tax information was obtained from employer by hacker who used information to file fraudulent tax returns).

Electronic copy available at: https://ssrn.com/abstract=3533517

may be deemed impermissible and therefore voidable at the option of one party or void as a matter of law. Particularly in the case of contracts that violate public policy, an affected party may have the option of voiding a contract, and in some circumstances, the law simply may render the contract void and therefore unenforceable. Both conditions are not currently possible under existing smart contract technology.

Contracts may be void or voidable for a number of reasons. While these vary by jurisdiction and include a wide variety of options, those relevant to this analysis all share the common characteristic that they involve conditions not deemed illegitimate by an enforcing authority before execution and where a court may or must order the contract unenforceable after execution.[75] Common examples include incapacity, duress, and violation of public policy.

Coerced agreements, for example may be voided by the person who was under duress.[76] Similarly, a party induced to enter an agreement by fraudulent misrepresentation "may instead elect to avoid the transaction and obtain restitution."[77] If the subject of the agreement is illegal[78] or violates public policy, the common law provides courts with the authority to set these deals aside and choose not to recognize them as legally enforceable contracts.[79]

Traditional contract doctrine requires that such transactions, once agreed upon, may be nullified by the courts.[80] For example, if Deborah and David enter into a common law contract to transfer a fixed amount of bitcoin, a court could intervene for a host of reasons. If David offered Deborah bitcoin in exchange for committing a murder, a court will not enforce the contract regardless of the parties' desires.[81] If David were a child, he could elect to avoid the contract.[82] If Deborah pointed a gun at David's head to force the transfer, a court could nullify the pact.[83] If David paid

[75] Often times this is after execution but before settlement, however this is not always the case. Those circumstances which are post-execution/pre-settlement are most salient to this analysis, because those are precisely the types of circumstances which are most problematic for existing smart contract technologies to address.

[76] *See* RESTATEMENT (SECOND) OF CONTRACTS §§ 174–175 (AM. L. INST. 1981).

[77] PERILLO, *supra* note 6, § 9.13, at 307.

[78] *See, e.g.*, Bassidji v. Goe, 413 F.3d 928 (9th Cir. 2005).

[79] *See* RESTATEMENT (SECOND) OF CONTRACTS § 178 (AM. L. INST. 1981).

[80] PERILLO, *supra* note 6, § 9.13, at 307.

[81] *See id.* § 22.1, at 762 ("A contract guarantying performance of an illegal act is itself illegal." (citations omitted)); *see also id.* § 22.1, at 763–64 ("As a general rule an illegal bargain is unenforceable and, often void. . . . The Restatement (Second) rejects consideration analysis of contracts against public policy. Under its analysis, A's promise to murder X is indeed consideration for B's promise to pay A $10,000. B's promise is unenforceable, not because of the lack of consideration, but because it is illegal." (citations omitted)).

[82] *See id.* § 8.1, at 259 ("There are certain classes of persons whose contractual capacity is limited. Their agreements are either void, or more often, voidable. These classes include infants and persons suffering from mental infirmity.").

[83] *See id.* § 9.1, at 285–86 ("Even though parties who have contractual capacity have expressed mutual assent and their agreement is supported by consideration or one of its equivalents, the agreement may be void, voidable, or reformable because it is contaminated by duress, undue influence, misrepresentation, mistake, or unconscionability. . . . Today the

Electronic copy available at: https://ssrn.com/abstract=3533517

Deborah for her silence to hide sexual misconduct, many states would consider that agreement to be void for violating public policy.[84] And if Deborah fraudulently induced David to agree to the transaction, the agreement could be judicially undone.[85]

In the case of smart contracts, however, all of these fact patterns could not accommodate judicial remediation. Once committed to the blockchain, a smart contract *will* execute after the triggering conditions in its executable code are satisfied.[86] There are two relevant categories of triggering conditions here: (1) conditions which do not depend on an outside Oracle; and (2) conditions which do depend on information from an outside Oracle. The vast majority of smart-contract-based agreements currently contemplated by modern dApps are of the latter variety.[87]

Smart contracts that have triggering conditions for settlement that do not depend on an outside Oracle are the most straightforward example of the failure of existing technologies to implement the requirements of common law contract doctrine. Consider, for example, a simple escrow agreement implemented via an Ethereum smart contract. Under the terms of this hypothetical agreement ("A1"), David would place 1 BTC[88] in escrow pending completion of a separate agreement ("A2") between Deborah and David such as a real estate agreement. In the event that A2 is finalized, executed, and committed to the Ethereum blockchain, the executable code of A1 would recognize some predetermined aspect of the code of A2 and trigger the release of funds from escrow to Deborah. However, if A2 is not finalized, executed, and committed to the blockchain by the time agreed to in A1, that bitcoin would be returned back to David. *There are no other possibilities*.[89]

general rule is that any wrongful act or threat which overcomes the free will of a party constitutes duress." (citations omitted)).

[84] *See id.* § 22.1, at 762 ("Public policy has been the announced rationale for striking down contracts or contract clauses on the grounds of immorality, lobbying, unconscionability, stock redemption, economic policy, unprofessional conduct, obstruction of justice, paternalism, ultra vires, defrauding of creditors, parental deals that prejudice their children's rights to support, and diverse other criteria.").

[85] *See id.* § 9.22, at 323 ("In the great majority of cases, actional misrepresentation renders a transaction voidable rather than void. These are cases of fraud in the inducement." (citations omitted)).

[86] Strictly speaking, there are conditions which might cause execution to fail. However, generally speaking, none of these conditions are within the control of a legal authority like a court (rather, they generally involve wide-spread system failures or similar corner cases).

[87] Les Wilkinson & Curtis Capeling, *How to Understand Blockchain*, ASS'N CORP. COUNSEL DOCKET, Sept. 2018, at 66, 69 ("Very often, smart contracts use 'oracles' to provide off-chain information (such as proof of payment or performance, or data from devices in the Internet of Things) necessary to the execution of a smart contract.").

[88] This is approximately $11,900 USD as of August 18, 2020 according to Coinbase. COINBASE, https://www.coinbase.com/price (last visited Aug. 18, 2020).

[89] It is important to note, of course, that contracts *could* be coded in other ways, but generally speaking, cryptocurrency smart contracts currently are limited essentially to two outcomes: transfer settlement or transfer reversal.

Electronic copy available at: https://ssrn.com/abstract=3533517

What if, however, it turns out that in the process of creating A2, Deborah attempted to coerce David by holding a gun to his head and demanding favorable terms? Not only would A2 be irreversible once committed to the blockchain, but A1 would also be triggered and settle. A court would have no ability to intervene.

If the execution depends on information supplied by an outside Oracle, it is possible (although certainly not guaranteed) that a court might be able to intervene. Such intervention would depend, however, on the court's power to affect the information supplied by that Oracle before the relevant time expires and would also depend on whether the terms of the contract provide for an appropriate judicial remedy. The court's redress options may depend on, but are not limited as a matter of law to, the terms of the contract. For example, a contract may not specify liquidated or other financial damages, but a court nonetheless may order them. Conversely, a court may find a liquidated damages provision unenforceable because it is unreasonable, the product of unequal bargaining power, or coercive action by a party. In that case, a court might order money damages in an amount different than that specified in the contract, or (less commonly) a court might award different forms of damages.

Much of our analysis might suggest the conclusion that current implementations of smart contracts are not contracts as a matter of legal doctrine. While some commentators have taken doctrinal positions on this matter, including that future implementations cannot resolve these shortcomings,[90] we argue that the doctrinal question is better left for another day when the underlying technology is more mature, and its inherent capabilities and limitations are better understood. However, this does not mean that the limitations of current smart contract technology are trivial. Quite the contrary, as discussed in Part II, those limitations are distinct from traditional written contracts with several critical differences and important implications for parties to consider when deciding whether or not a smart contract is appropriate to govern their agreement.

C. The Benefits of Bot Contracts

Common law contract remedies developed to inspire trust in transactions between people who did not know each other.[91] If one party broke a contractual promise, the other can trust that a court might intervene. Of course, the ability to

[90] *See* Schuster, *supra* note 70.

[91] *See* RESTATEMENT (SECOND) OF CONTRACTS ch. 16 intro. note (AM. L. INST. 1981) ("The traditional goal of the law of contract remedies has not been compulsion of the promisor to perform his promise but compensation of the promisee for the loss resulting from breach."); *see also* G. Richard Shell, *Opportunism and Trust in the Negotiation of Commercial Contracts: Toward a New Cause of Action*, 44 VAND. L. REV. 221, 222–23 (1991) ("The law of contract helps to diminish the danger of opportunism by providing assurance to those performing first that their contracting partners can be held accountable if they renege. Accountability reduces the risk of entering business transactions and facilitates an atmosphere of confidence conducive to exchange."); *id.* at 225 ("To sustain trust and enhance the chances for reciprocity, parties also must be willing and able to deter those who might act opportunistically after a trusting relationship is underway. . . . [T]here is a need to devise mechanisms that deter people who are tempted to violate trust. Legal recourse for victims of opportunistic conduct is one possible remedy for bolstering the cooperative process.").

Electronic copy available at: https://ssrn.com/abstract=3533517

obtain relief based on breach or void an unlawful deal is contingent on being able to hire a lawyer.[92] And there is a substantial literature on how access to counsel and courts is a luxury to which many do not have access.[93] The transactions known as smart contracts avoid the need to put trust in another party or the judicial system. Software is designed to ensure performance. Resources are not needed to ensure accountability or performance. The technology performs the obligations automatically, taking the uncertainty of human performance accountability out of the equation. The technology alleviates the need to trust any particular person to ensure contract settlement once the terms have been satisfied. It is an ideal solution for two parties who both want to unlock a car or conduct a transaction with the understanding that the transaction will be fixed. Importantly, these technologies do not remove all ambiguity or obviate all need for trust.[94] One must still place trust in the technology and the community's attributions of meaning attached to various terms in the code.[95] In short, current smart contracts are ideal for circumstances where automatic self-settlement of contracts is desired, where the terms and outcomes of the contract are readily predictable in advance, where customary market terms will likely govern transactions, and where the parties are comfortable waiving traditional rights and defenses associated with common law contracts, such as duress, fraud, and incapacity.

D. Future Developments

It is important to recognize that computing and information technology constantly evolves. With the exception of certain subsets of theoretical modeling,[96] "truth" and "proof" about the "state of technology" are subject to a very large qualification *given the state of technology at the time the assertion is made*. Recognizing this critical distinction, which is similar to the qualification "under current law," this Section identifies some of the assumptions upon which this Article rests and examines probable future technological developments which may impact its conclusions.

First, we begin with the assumptions on which this Article relies. We describe the state of commonly used blockchain technologies as of approximately

[92] *See* Deborah L. Rhode, *Whatever Happened to Access to Justice?*, 42 LOY. L.A. L. REV. 869, 869 (2009) ("Litigants who remain unrepresented are less likely to obtain a fair outcome in court.").

[93] *See generally, e.g.*, Jessica K. Steinberg, *Demand Side Reform in the Poor People's Court*, 47 CONN. L. REV. 741 (2015); Dina E. Fein, *Access to Justice: A Call for Progress*, 39 W. NEW ENG. L. REV. 211 (2017); Columbia L. Sch. Hum. Rts. Clinic, *Access to Justice: Ensuring Meaningful Access to Counsel in Civil Cases*, 64 SYRACUSE L. REV. 409 (2014).

[94] *See* Grimmelman, *supra* note 63, at 20.

[95] *See id.*

[96] Generally, these are not relevant to this discussion. But even to the extent some exceptions are, those are subject to the general qualifications of mathematics and physics—i.e., "that we are performing base-10 arithmetic," "that we are operating in an environment like that on the surface of the Earth," and in the extreme, "that we are operating in a place in the Universe where traditional principles of Newtonian physics apply." These examples serve only to illustrate that certain assumptions *always* apply, which is indeed the point to which this Section is responsive.

Electronic copy available at: https://ssrn.com/abstract=3533517

early 2019. We focus on technology currently in widespread use for business and other organizational applications. These qualifications generally describe the technologies with which much blockchain-related legal scholarship engages,[97] and thus, we follow in that tradition with only the minor modifications suggested in Part II.

We recognize that many of our conclusions may be altered, possibly significantly, as the state of the technology changes. This Article interrogates the degree to which contract law's promises are fulfilled by current blockchain technologies due to the technology's inability to act according to the common understanding of its name. We encourage future authors to apply it to new and developing technologies. We do not claim that smart contracts will *never* be able to act like common law contracts. Such a claim would be rank hubris. Rather, we claim that they *currently* fail to do so and note the following potential future developments which may impact that analysis.

Perhaps the most critical aspect of the current state of smart contract technology is its inherent (generally) irrevocable self-executing nature.[98] Early versions of such technology, most notably early iterations of Ethereum, did not construct their governance structures enabling anything other than self-executing smart contracts. The Turing-complete nature of Ethereum technically could insert a third-party arbiter at a specific, predefined time prior to final resolution (in the form of an Oracle). However, such an intervention would be limited to predefined time(s), which limits the ability of the system to provide remedies at all the times such remedies would be available under common law contracts. Additionally, such an intervention protocol would necessitate an affirmative response from the arbiter or a "timeout" period,[99] either of which would impose nontrivial transaction costs at least in the form of delay.

There is nothing inherent in the science of DVTs, however, preventing the implementation of a third-party neutral arbiter into the smart contract resolution process. Consider, for example, a modification of the Ethereum platform whereby smart contract resolution resulted in the transfer of a certain amount of currency, but where that currency was "subject to recall" for a specific period. If a contract defense were raised during that period, the funds could be "recalled" by a designated arbiter into escrow pending the outcome of appropriate judicial or other legal resolution.[100]

[97] *See generally*, *e.g.*, KROLL ET AL., *supra* note 20; Turpin, *supra* note 38; Werbach & Cornell, *supra* note 31; Schuster, *supra* note 71 (Although Schuster does attempt to look forward at developments, we feel his analysis is incomplete.).

[98] Technically, irrevocability depends on the inability of anyone (or even a few) parties to control the consensus mechanism. As a practical matter—and as discussed in Part II—it is not realistic that any such attacks would be implemented against a given contract. *See also supra* note 16.

[99] A period during which the transaction was essentially "frozen," after satisfaction of performance but before settlement of funds, during which dispute resolution could occur.

[100] Such an approach is critically distinct from attempting to implement such mechanisms within each current smart contract because common law contract doctrine requires those defenses *always* be available, i.e., one can't "contract around them."

Electronic copy available at: https://ssrn.com/abstract=3533517

It is quite plausible for such a feature to be implemented into future smart contract platforms.

A related but distinct technological advance that could alter our analysis would be the implementation of a method for interactive contract enforcement and adjudication. Such an implementation is far closer to the current state of the technology, and indeed, some blockchain industry projects have been working toward such a concept.[101] Interactive enforcement and adjudication is related to the creation of third-parties capable of interrupting execution to enforce defenses. Indeed, one (albeit incomplete) method of implementing "execution interrupts" is through interactive enforcement and adjudication. However, it is important to distinguish between these concepts, as even interactive adjudication and enforcement based on execution "interruption" depends on predetermined scripted conditions lacking the independent, human-driven qualitative judgment associated with traditional judicial fora.

Interactive adjudication and enforcement, usually at a specified point in contract execution, allows for external input including an interactive process to determine whether or not a smart contract should resolve. This can *partially* implement "execution interrupts" but can only do so at a prespecified time and under prespecified terms.[102] By contrast, implementing "execution interrupts" does not create a process for qualitative evaluation of whether a contract *should* resolve, so much as creating a procedure to interrupt or reverse resolution in the event a certain type of event, i.e., a contract defense, has occurred.

The importance of interactive adjudication and enforcement becomes clear in the context of contracts whose fulfillment depends on the evaluation of some qualitative term, such as artistic or creative satisfaction. Determination of such fulfillment depends on qualitative analysis currently outside the capabilities of smart contract platforms, except through the use of input from Oracles. And indeed, some smart contracts currently use Oracles for this purpose. Oracles standing alone, however, are incomplete because they do not embed the type of deliberative process guaranteed by judicial resolution of common law contracts. Even binding arbitration clauses in common law contracts may result in a process allowing each party to proffer facts, argue their position, and respond to arguments and evidence submitted by the opposition.[103] Current smart contract platforms do not implement these

Implementing such things in each individual smart contract places them under the control of the contracting parties, which is inconsistent with the requirements of common law contracts. By contrast, implementing such third-party resolution provisions through the *system* not only would apply them to all parties, but critically would take such decisions out of the hands of the contracting parties, consistent with the requirements of common law contract doctrine.

[101] *See, e.g.*, *Overview*, OPENLAW, https://docs.openlaw.io/ (last visited Sept. 22, 2020).

[102] And thus, it does *not* implement all the contract law defenses discussed *supra* Section III.B.

[103] *See, e.g.*, Noohi v. Toll Bros., Inc., 708 F.3d 599, 614 (4th Cir. 2013) (holding that an arbitration clause binding only one party was unenforceable for lack of mutual consideration).

Electronic copy available at: https://ssrn.com/abstract=3533517

abilities. However, there is nothing scientifically preventing them from doing so in the future.[104]

CONCLUSION

Current smart contract technology simply cannot implement all the aspects of legal contracts under common law doctrine. We do not mean to say that technological development cannot achieve those aspects in the future. It is too early for such determinations to be made. Likewise, none of the points are intended to indicate that we do not find smart contract transactions to be of considerable value. Three great values of these transactions are that they remove risk, reduce uncertainty, and eliminate the need for court intervention. However, these transactions are not contracts in the common law sense of the term. And they are not smart. Once set into motion, they will execute once the triggering conditions are satisfied. For all of these reasons, they are better conceived as automated rather than smart, and fixed rather than alterable by judicial remediation. They are Bot Contracts.

[104] It is worth noting that such processes may impose high transaction costs and obviate many of the economic benefits of smart contract-style agreements over traditional common law contracts. It is also worth noting that some scholars have argued that these problems *cannot* be overcome. *See* Schuster, *supra* note 71, at 26–29. Such arguments generally fail, however, because they assume certain static aspects of the system, and such assumptions are provably untrue. Oracles, for example, are not (as Schuster claims) forced to a binary choice of either conveying information automatically or becoming central authorities which mediate the application of judicial judgments. *Id.* at 27. In one simple counterexample, code could be developed in a smart contract to take external input and determine whether or not to release code from escrow during a "hold" period. Or, alternatively, code could include reversibility functions which last for the durations of the applicable statute of limitations. The fact that a recipient spends the money and thus may make themselves lack the means to reverse the transaction is no different than a defendant in a contract case who makes themselves "judgment-proof" in the traditional common law sense.

Electronic copy available at: https://ssrn.com/abstract=3533517