

Lee Bacon / George Bazinas

## «Smart Contracts»: The Next Big Battleground?

---

With the advent of blockchain as a decentralised database, with the ability to store a public registry of assets and transactions across a shared, trusted, peer-to-peer network, and the sophistication of software code as a form of communicating information and automating complex instructions, blockchain has the capacity to significantly disrupt and transform the legal and business landscape. In this article we consider the extent to which «smart contracts», agreements reduced or replicated in software code whose execution is both automatable and enforceable, can replace traditional legal contracts.

---

Category: Essay  
Region: United Kingdom  
Field of law: LegalTech

Citation: Lee Bacon / George Bazinas, «Smart Contracts»: The Next Big Battleground?, in: Jusletter IT 18 May 2017

## Contents

1. Brief Introduction into Smart Contracts
2. Usefulness of Smart Contracts
3. The Fundamental Problem of Immutability
4. The Answer of Enforceability through ADR

[Rz 1] The advancement of blockchain technology as a decentralised database with the ability to store a public registry of assets and transactions across a shared, trusted peer-to-peer network is lauded as one of the most significant technological innovations since the internet. Coupled with the sophistication of software code as a form of communicating information and automating complex instructions, blockchain technology provides a new means of recording information and facilitating the exchange of value in the global economy in a decentralised and immutable way.

[Rz 2] At the centre of this battleground lie so-called «smart contracts», which are essentially agreements reduced to or replicated in software code whose execution is both automatable and enforceable. In this article we consider the extent to which smart contracts can replace traditional contracts. Fundamentally, we believe this rests not only on their operational and denotational parameters, but also on their usefulness as a form of contracting language.

### 1. Brief Introduction into Smart Contracts

[Rz 3] Given the lack of a clear consensus on the definition of the term «smart contract», it is necessary to provide an overview of the two different ways that the term «smart contract» is commonly used in the business world. Broadly, the first is used to specify software code that is typically stored, verified and executed on a blockchain («*smart contract code*»), while the second focuses on legally enforceable contracts partially expressed and/or executed in code («*smart legal contracts*»).

[Rz 4] Importantly, smart legal contracts differ from smart contract code in terms of *what* is being enforced; the former involves the enforcement of complex legal rights and obligations while the latter may simply involve the execution of the code. When speaking about the transformational and disruptive capacity of smart contracts, therefore, we typically refer to smart contract code or business logic which can be programmed, verified, executed, and enforced automatically under conditions set in advance.

[Rz 5] One other important distinction to make in this context is between syntactic (or operational) information, which describes the rules governing the relationship between symbols and coded language, and semantic (or denotational) information, which describes the meaning attributed to such symbols or the intent behind the coded language. It is our view that semantic information, like the notion of «good faith», cannot yet be translated into code. Undoubtedly, this poses a significant obstacle to the practical implementation and widespread adoption of smart contracts.

### 2. Usefulness of Smart Contracts

[Rz 6] One of the key advantages of smart contracts on a blockchain is that transactions are recorded accurately and consistently in a distributable and shared way that allows every party (or

node) in a network to verify the accuracy of the central ledger by reference to their own copies. This system embeds trust in the contractual relationship, enabling multiple parties to transact with each other with the certainty that a transaction will be performed as agreed. Linked to this is the transparency of source code, which allows any party to access the written code online and independently verify its functionality.

[Rz 7] Second, aside from the inherent limitation to capture legal concepts and intent, blockchain based smart contracts offer not only more flexibility in the implementation of decentralised digital asset transfers, but also the capability to perform a range of functions that traditional contracts cannot. For example, in a traditional sale of goods, smart contracts can automate both the performance and payment element of a contractual agreement by reference to an external set of dynamic or live conditions, such as the fluctuation of the market price of goods, the exchange rate, or even the weather. Smart contracts have the capacity to streamline business processes and transform traditional business models in industries that strongly rely on such external dependencies, like finance, international trade, and insurance, something which we are already beginning to see. With the added capability offered by the internet of things, smart contracts are poised to radically transform and disrupt business as we know it.

[Rz 8] Related to the above, smart contracts are less likely to result in a mismatch or breach of the terms in an agreement. In a traditional contract, there can be a mismatch where there is a mutual misunderstanding of the initial terms, confusion due to multiple forms or versions of the original terms, or a disagreement with what actually happened in the external dependencies. With a smart contract, there is only one set of contractual terms, written in software code, less verbose than legalese, and agreed upon in advance by reference to external dependencies that are fed in via a mutually agreed feed (or so-called oracles). It stands that the smart contract will exist on the blockchain and run when an event occurs or when the agreement expires. While this does not offer a utopian dispute-less context, the scope for disputes would be greatly reduced.

[Rz 9] Finally, smart contracts on an immutable blockchain provide the added certainty that an agreement will perform as agreed, in strict accordance with the code. Once a smart contract is programmed, it can no longer be changed, unless parties build in an express provision that allows for performance or execution of the contract to be halted on a set of mutually agreed conditions. Although an integral feature of the functionality and usefulness of blockchain, as discussed below, immutability also poses the most significant challenge for the practical implementation and adoption of smart contracts at the level of industry.

### **3. The Fundamental Problem of Immutability**

[Rz 10] As displayed by «The DAO attack», which involved the exploitation of a flaw in the code of a smart contract on the Ethereum Blockchain, smart contracts – when immutably embedded in a blockchain – cannot be changed or updated. This means that defects and vulnerabilities in the code cannot be fixed and code is effectively deprived of the benefit of being subject to a process of refinement and testing to allow it to reach «maturity».

[Rz 11] This problem is further expounded by the difficulty, or near impossibility, of writing large pieces of code free of any bugs, as well as the inability of code to capture the semantic intent of the developer or curator. This is particularly visible in the case of The DAO attack given the inability of code to process the «good faith» directive written on The DAO's home page (i.e. the

recital that provided for «doing no harm»), which arguably could have prevented The DAO attacker from withdrawing funds from The DAO, had it been possible to functionally execute such a directive. Ironically, although the code did not have the outcome intended by the investors or developers of the fund, it executed the instructions of the smart contract faithfully and precisely as programmed.

[Rz 12] The immutability of the Ethereum Blockchain, this incapability to fix vulnerabilities in the written code, coupled with the inherent inability of processing semantic information or capturing the real intent behind The DAO smart contract, is what ultimately led to the logical impasse that resulted in the decision to change (or hard fork) the Ethereum Blockchain protocol, effectively reversing The DAO transactions.

[Rz 13] The hard fork drew criticism from both within and outside the Ethereum community and was labeled as a unilateral act of governance by the Ethereum Foundation that ran contrary to its alleged founding values of democratisation and decentralisation. In the widely publicised aftermath, adherents of the strict «code is law» doctrine resisted the change and condemned the act by splitting into a separate, parallel blockchain, called Ethereum Classic. As might have been expected, the hard fork was then followed by two successive forks, aimed at addressing ongoing attacks on the network that slowed down transactions and smart contracts, culminating in what were effectively three forks in the space of four months. While the Ethereum community appears to have settled on hard forks as a regular way of fixing technical problems, it is questionable whether hard forks in this manner offer a sustainable or even satisfactory means of governance for Ethereum and other blockchains at large.

#### **4. The Answer of Enforceability through ADR**

[Rz 14] In a context where the certainty and immutability of contract code is considered one of the chief attributes of blockchain based smart contracts, the repeated interventions by Ethereum pose a serious question on the long-term viability of the blockchain project. The hard fork potential invites the floodgates argument – if Ethereum intervenes to roll back transactions or sequester funds in this case, what stops stakeholders from requesting and Ethereum from reversing transactions and carrying out successive corrective forks in future smart contract failures?

[Rz 15] While a smart contract may be functionally executable to the extent that the transfer of assets and payment is all that is required, in situations where the code does not perform as intended by the parties or there is some form of disagreement, there is currently no formal redress system to resolve such issues or disputes. To this end, Alternative Dispute Resolution («ADR») provides parties with the formal means to resolve disputes in a private setting and without resorting to courts. This offers a two-fold advantage over the informal and unilateral approach taken by Ethereum. First, it transfers legal authority to an independent and experienced arbitrator, who uncontroversially is qualified to hear and rule on disputes. Second, it elevates computer code to binding contractual provisions and provides a legal platform for recognising and enforcing legal rights in a smart contract.

[Rz 16] In this respect, not only are arbitrators better placed to consider principles of contract law to determine what the parties have agreed, but they also have the authority to render a legal judgment that would be enforceable in a court of law. Naturally, given the lack of representative capacity of constituents of the blockchain (i.e. developers, creators, or investors) and the lack

of recognized legal form of distributed autonomous organisations («DAOs»), which are organisations that are run through rules encoded as smart contracts, ADR provisions would need to be tailored to suit the nuanced requirements of such transactions.

[Rz 17] Notwithstanding the challenge of developing such a discrete and adaptable set of smart contract provisions, ADR offers an important opportunity to test blockchain technology disputes and develop an effective system of governance that balances the values of blockchain immutability with the need for certainty in contracts. Although we are still a long way from smart contracts entirely replacing traditional legal contracts, we have already begun moving in that direction and it seems clear that the businesses that keep apace with technological innovation in this rapidly evolving area will gain a competitive advantage.

---

LEE BACON, Partner at Clyde & Co LLP in London, specialises in insurance and reinsurance as well as managing litigation, arbitration and mediation. [lee.bacon@clydeco.com](mailto:lee.bacon@clydeco.com).

GEORGE S. BAZINAS, LL.M., Trainee Solicitor at Clyde & Co LLP in London. [george.bazinas@clydeco.com](mailto:george.bazinas@clydeco.com).