

Smart Contracts – Dumb Idea

Kieron O'Hara

*Web and Internet Science Group
Electronics and Computer Science
University of Southampton
Highfield
Southampton SO17 1BJ
United Kingdom
kmo@ecs.soton.ac.uk*

Accepted version: *IEEE Internet Computing*, 21(2), 97-101 (2017)

In Summer 2016, the world of distributed public ledgers, blockchains, cryptocurrencies and trustless trust was agog as a \$55 million hack unfolded on the Ethereum platform. Like a slow-motion car crash, it piqued the interest for a few days, taking everyone's minds off the impending Brexit vote and the unedifying US election campaign. In the end, the hack was thwarted, but the final resting place of the money is far less important than the issues it raised.

Smart contracts

Techies like to develop solutions for problems that no-one ever noticed.¹ The rationale, in so far as there is one, is that a messy, scruffy real world needs to be tidied up by code, because tidiness is a virtue.

In *The Laws* (written about 350BC), Plato wrote a little about contracts in a way that implied that they were a well-understood and familiar part of contemporary civic life in Athens. He talked about what might happen if a man (probably only men could undertake contracts then) reneged on a contract, and who should arbitrate between the disputing parties. He also suggested that not all contracts were valid – for example, where someone contracted to do something illegal, where one of the parties consented under pressure, or where the failure to carry out the contract was not the fault of the party concerned. He didn't go into much detail, but enough to indicate that the Ancient Greeks had organised their affairs via contracts quite well for some time thank you very much and didn't require the gods (not even in Silicon Valley) to adjudicate.

Fast forward to the 1990s, when it began to be argued that e-commerce had created a need for contracts to be smarter.² The idea of a smart contract is surprisingly difficult to pin down, but a recent definition was “a mechanism involving digital assets and two or more parties, where some or all of the parties put assets in and assets are automatically redistributed among those parties according to a formula based on

¹ Evgeny Morozov, *To Save Everything, Click Here: Technology, Solutionism, and the Urge to Fix Problems that Don't Exist*, Philadelphia: Perseus, 2013.

² Nick Szabo, 'Formalizing and securing relationships on public networks', *First Monday*, 2(9), 1997, <http://firstmonday.org/ojs/index.php/fm/article/view/548>.

certain data that is not known at the time the contract is initiated.”³ By being party to a contract, agents agree to perform (or not perform) certain tasks, and the smart contract would reduce uncertainty and transaction costs. In the ideal, the mechanism (program) of the smart contract would implement the contract. Neither side could renege, thanks to the remorseless logic-crunching of the machine, whose algorithm would execute, verify and enforce itself.

The applications no doubt go well beyond contract – any kind of agreement that requires the parties take on risk during coordinated movement of assets could be rebooted into the twenty-first century with such mechanisms (including criminal schemes⁴). Trust is unnecessary, because the execution of the agreement is no longer separate from the agreement itself. The code *is* the contract. Its execution is the undertaking.

This should give us pause. Part of the point of contracts, as with other types of constraining commitment such as promises and vows, is to increase trust and spread it more widely than local social mechanisms such as those based on acquaintance, kin or tribe. Contracts are part of the panoply of means by which we globalise trust. In a society with the rule of law and in which contracts are generally respected, cooperation is easier and less risky (and therefore more common). Odd therefore to position a form of contract as reducing the need for trust. Let’s park this thought for now – we will return to it.

Building on smart contracts

Smart contracts are an old (in digital years) idea, but their implementation became practical with another more recent technology that does not trust trust: blockchains. The terms of the contract and the assets to be transferred can be arithmetically coded into the open-source, consensus-based, timestamped chain, and its execution can be verified independently. Neither party can independently modify the code, and so a blockchain-enabled smart contract will simply chug on to its conclusion, no matter how long that takes or indeed whether either party changes its mind about the contract itself.

Now that smart contracts have an implementation mechanism, we can envisage how to build on them. For instance, an organisation is basically a group of agents cooperating in the pursuit of some kind of goal or type of practice, and many organisations – private enterprise firms most obviously – are structured using contracts, both to constrain the roles of owners, members/employees and outside stakeholders (e.g. customers), and to assert property rights over buildings and other assets. Make those contracts smart, and you get an organisation controlled not by a hierarchy (SO twentieth century, dahling!) of (irrational) humans using (untidy) law, but a more autonomous peer network interacting via the rational, inexorable blockchain protocol.

(At least) one platform, Ethereum (<https://www.ethereum.org/>), has been developed with an eye to supporting smart contracts. A rival cryptocurrency to Bitcoin, the ether,

³ Vitalik Buterin, ‘DAOs, DACs, DAs and more: an incomplete terminology guide’, *Ethereum Blog*, 6th May, 2014, <https://blog.ethereum.org/2014/05/06/daos-dacs-das-and-more-an-incomplete-terminology-guide/>.

⁴ Ari Juels, Ahmed Kosba & Elaine Shi, ‘The ring of Gyges: using smart contracts for crime’, *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, 283-295, 2016, <http://initc3.org/files/Gyges.pdf>.

underpins the mechanism, to encode the assets and pay for services and fees, but on top of this a more sophisticated scripting language facilitates a wider set of functions. Ethereum, at one stage apparently worth a cool billion dollars, has been growing at Bitcoin's expense as the latter struggles with expanding network capacity.⁵

The Holy Grail is an autonomous organisation governed by smart contracts whose operation is as far removed as possible from the day-to-day input of its members or owners – a Decentralised Autonomous Organisation (DAO). The edges of the definition turn out to be fuzzy when you look at the details, but the idea is clear enough; as it is decentralised it lacks a bottleneck or single point of failure, while as it is autonomous it runs itself. A DAO is not too different in principle from a complex multinational company; firms are often owned by other firms (which count as legal persons), and it can be hard to work out who owns and is responsible for conglomerates like these. Some economies, such as Italy's or South Korea's, contain deep overlapping networks of cross-ownership. Can the actions of a single company within those economies be identified easily as the responsibility of a particular person or group of people? Not always. In that context, a DAO may seem the next logical step.

DAO ker-ching!!

The most prominent DAO called itself, with impressive hubris,⁶ The DAO. The DAO was (note the past tense) a capital investment fund run on the basis of peer-to-peer smart contracts, as well as a home for early adopters keen to show the viability of the concept. Investors bought ether coins to join the fund, in which they received a vote proportional to their investment. A candidate for funding would put forward a business plan together with a smart contract to define its relationship with The DAO; investors would vote on whether to support the candidacy. A 'no' would mean there would be no relationship; a democratic 'yes' would trigger the smart contract and under the rules that it set, funds would flow.⁷ This, the largest crowdfunding campaign in history, raised \$150 million in May 2016.

By June, it had collapsed. Although there had been scepticism and some cautionary voices amidst the hype, they missed the proximate cause. Some had worried that investment decisions, properly taken, were time-consuming, and so the number of votes cast for each decision might be too small to leverage the wisdom of crowds. Withdrawing uncommitted money was simple to do, and so some commentators were concerned that the \$150 million might disappear overnight. Still others wondered how the world's financial system would cope with a company that was not registered in any state, and which had no employees. As an article in *The Economist* opined, in the world of cryptocurrencies, faith and rationality go together like yin and yang.⁸

Yet the problem was even more obvious than any of these difficulties. A bug in the code was exploited. The system depended on smart contracts, and if these were not

⁵ Timothy B. Lee, 'Ethereum, explained: why Bitcoin's stranger cousin is now worth \$1 billion', *Vox*, 24th May, 2016, <http://www.vox.com/2016/5/24/11718436/ethereum-the-dao-bitcoin>.

⁶ Harriet Green, 'Introducing the DAO: the organisation that will kill corporations', *City A.M.*, 3rd May, 2016, <http://www.cityam.com/240198/introducing-the-dao-the-organisation-that-will-kill-corporations>, Adam Cleary, 'Here is how the DAO will soon become the greatest threat banks have ever faced', *Frontera*, 25th May, 2016, <https://fronteranews.com/news/dao-will-soon-become-greatest-threat-banks-ever-face/>.

⁷ 'The DAO of accrue', *The Economist*, 21st May, 2016.

⁸ 'The DAO of accrue', *The Economist*, 21st May, 2016.

secure (and research has located vulnerabilities in Ethereum's code⁹), there would be nothing standing between hackers and 150 million smackeroonies. Less than a month after the flurry of publicity, \$55 million disappeared in the general direction of who knows who, via a replay attack in which the same transaction was repeated over and again. Fortunately (depending on your point of view – see below) the hack required the money to be siphoned off into a subsidiary bank account where it sat for long enough for Ethereum's coders to devise and implement a hard fork to recover the cash¹⁰ and restore it to the investors (who immediately and wisely took it out of the system altogether, no doubt breathing heavily and patting sweat from their brows).

Yet was this ethical, or principled? Recall, The DAO was premised on smart contracts, whereby the code is the contract. The contract could not therefore be rescinded, and trust in the system was not needed – such was the rhetoric. Yet in the face of a loss which used the code as written, the smart contracts were indeed rewritten. A hard fork bifurcates the blockchain – it is a change in the rules for validating blocks that are the basis for consensus. After the rule changes, the chain diverges into two incompatible chains, one of which follows the new rules and the other which clings to the old ways (and, it is hoped, withers away). The relationship between these two branches – and the asset allocations they encode – is non-trivial.¹¹ Many in the Ethereum community thought this played fast and loose with the ideology behind their innovation (which it does), and preferred to continue on the old fork (christened Classic Ethereum).

The problem is clear: if the code is the contract, then whatever the hackers did was permitted within the contract. The facts that all code is buggy, and that the Ethereum coders clearly did not intend to license replay attacks, are neither here nor there. It was Ethereum which broke the contract, not the hacker, because The DAO, which held about one ether in seven in circulation, was deemed too big to fail. However, as we discovered in 2008, just because an entity is judged too big to fail, that does not mean that it won't fail.

The rules is the rules, except where they isn't. As the great philosopher Mike Tyson once said, “everyone has a plan till they get punched in the mouth.” How stands trustless trust now?

Indeed, if the code is the contract, could the hacker even sue for his or her money back? Gosh, that would be a juicy case!

The human factor

If Ethereum can break its own unbreakable contracts, then the certainty that is supposed to be its USP disappears. Yet smart contracts were always highly inadequate types of contract.

Can you engineer humans out of contracts? Suppose a smart contract, in insurance for example, is to pay out after a certain event (say, an extreme weather event). If it is

⁹ Loi Luu, Duc-Hiep Chu, Hrishikesh Olickel, Prateek Saxena & Aquinas Hobor, 'Making smart contracts smarter', *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, 254-269, 2016, <http://www.comp.nus.edu.sg/~loiluu/papers/oyente.pdf>.

¹⁰ Vitalik Buterin, 'Hard fork completed', *Ethereum Blog*, 20th July, 2016, <https://blog.ethereum.org/2016/07/20/hard-fork-completed/>.

¹¹ Timon Rapp, 'How to deal with the Ethereum replay attack', *medium.com*, 17th July, 2016, <https://medium.com/@timonrapp/how-to-deal-with-the-ethereum-replay-attack-3fd44074a6d8#.vu14sdqyl>.

distributed across the blockchain, what or who is responsible for retrieving the data? Which copy of the contract activates the process? Whenever anything happens in the non-digital world as a result of the smart contract, there will surely need to be a human in the loop if only as a tiebreaker – the trusted third party turns out to be inevitable after all.

This is unsurprising, but there is a deeper point – the notion of a smart contract rests on a fundamental misunderstanding of what a contract is there to do. Contracts are not mechanisms to make things happen. They are social arrangements, voluntary constraints not unlike,¹² but not identical to,¹³ promises, backed by the machinery of law.

Contracts have a social function (which is why they have legal underpinning). They enable cooperation, and help spread habits of warranted trusting around an economy. Society benefits from regular and reliable exchanges of goods and services; it is hard to imagine the money economy functioning without contract (money itself is another type of promise). Many types of agreement receive support from rich networks of norms (of friendship, or kinship for instance), whereas parties in contracts often have very little in common other than the contract. Hence the trust-building function is key to the social value of the institution.¹⁴

Furthermore, the social good of contracts is promoted by the possibility of the contract being broken. That may seem paradoxical, and if the whole point of a contract was that contracted action A_1 be performed by agent X, then X's failure to perform it would be a moral outrage. Yet the law's remedies are not generally punitive, and do not enforce the performance of A_1 . In most jurisdictions all that other parties to the contract can expect to receive from X by way of damages is the expected value of A_1 . Furthermore, the law also expects the other parties to take steps themselves to minimise the damage caused by A's default.

The result of this is the optimisation of the social benefit of X's resources. If X can get better use of her resources by doing something other than A_1 , then she is able to, and is better off even after compensating the other parties to the contract (who are no worse off). This is called the doctrine of efficient breach.¹⁵

Won't this foster a culture of opportunism and betrayal? It hasn't yet; contracts are respected, and the law could always be changed if this ceased to be the case. The point made by supporters of efficient breach is that making the compensation equal to the losses following from the breach of contract encourages efficient breach while discouraging inefficient breach; any more is mere paternalism. Note also that even in this limited sense, contract still goes beyond a promise; a promisor neither opens

¹² Charles Fried, *Contract As Promise: A Theory of Contractual Obligation*, 2nd edition, Oxford: Oxford University Press, 2015.

¹³ Randy E. Barnett, 'Contract is not promise: contract is consent', *Suffolk University Law Review*, 45(3), 2012, Seana Valentine Shiffrin, 'The divergence of contract and promise', *Harvard Law Review*, 210, 2007.

¹⁴ Dori Kimel, 'Personal autonomy and change of mind in promise and in contract', Gregory Klass, George Letsas & Prince Saprai (eds.), *Philosophical Foundations of Contract Law*, Oxford: Oxford University Press, 2014, 96-115.

¹⁵ Richard A. Posner, *The Economic Analysis of Law*, 9th edition, New York: Wolters Kluwer Law & Business, 2014.

herself up to legal scrutiny, nor takes upon herself responsibility for correcting harms done by failure to deliver.¹⁶

Perhaps most important of all, contract has built into it the presumption that interpretation and flexibility will be needed, partly to deal with failures to agree on the meanings of particular commitments, partly because of the immense complexity of some contracts, for example governing major pieces of infrastructure, and partly because things change and both parties may want and expect the contractual terms to evolve over time. The courts can adjudicate here, and can also strike down unfair contracts, such as usurious loan rates or an unjustly one-sided employment arrangement.¹⁷ Contracts are also rarely in one direction; they generally involve reciprocity or exchange, and so have the additional complexity that brings.¹⁸

There is a balance between the words that each party signs up to (the textual interpretation), and what each party wants out of the agreement (the intentional interpretation). Naturally there are arguments for each interpretation, and the courts seek a balance. This is not a bug, as techies may think, but a feature. On the other hand, there is no way back from the smart contract (other than the hard fork, impractical as a general remedy for obvious reasons) if parties have misunderstood the specification of the code, if the code is badly-written, or if one party has been coerced or misled into taking on an unfair obligation.

There are, no doubt, several important roles that could be filled by smart contracts. In some places, the rule of law may be shaky, or courts may be congested. Routine or short-term commitments might be better served by algorithms than contracts, and one could easily imagine arrangements within an organisation being made using a blockchain. Intra-entity trust is a less pressing issue than inter-entity trust, and so the inflexibility characteristic of the smart contract is less likely to cause long-term problems if used to allocate resources within a single organisation. There is also more likely to be agreement about terms.

But smart contracts are dumb contracts, and the best contracts are fallible and human. In his influential book *Code*, Lawrence Lessig drew our attention to various means of constraining human behaviour – regulations, social norms, economic incentives and code or architecture.¹⁹ This important argument has resonated in many contexts, and has been a key axiom for twenty-first century digital politics.

However, it has also led to a dangerous fallacy. Just because one can imagine mechanisms of the different types being used to constrain behaviour, it does not follow that these mechanisms are interchangeable. It makes an enormous difference if we replace a legal constraint with software. In the first place, the law can be challenged, whereas in software the forbidden option is irreversibly greyed out and inaccessible. Secondly, the law is developed and administered transparently by our

¹⁶ Randy E. Barnett, 'Contract is not promise: contract is consent', *Suffolk University Law Review*, 45(3), 2012.

¹⁷ Seana Valentine Shiffrin, 'Paternalism, unconscionability doctrine and accommodation', *Philosophy and Public Law*, 29(3), 205-250, 2000.

¹⁸ Dori Kimel, 'Personal autonomy and change of mind in promise and in contract', Gregory Klass, George Letsas & Prince Saprai (eds.), *Philosophical Foundations of Contract Law*, Oxford: Oxford University Press, 2014, 96-115.

¹⁹ Lawrence Lessig, *Code and Other Laws of Cyberspace*, New York: Basic Books, 1999.

democratically-elected representatives and the courts; software development, even open source, is opaque, and concentrated in a small programming community, many of whom are in the pay of a few oligopolistic corporations directly accountable to no external party.

Third – most important from my point of view – we can break the law. There are consequences when we do, and the system would break down if we disobeyed it all the time. But the law is consistent with maximal liberty; it cannot compel obedience (though it can disincentivise it). Code, on the other hand, will not allow behaviour inconsistent with itself.

With smart contracts, this rules out the desirable economic gain of efficient breach. But in the wider context, the law's openness to breach allows many vital liberties, not least the civil disobedience that helped shape our civilised liberal order from Thoreau to Gandhi to King. The next time we consider replacing law with code to tidy up a scruffy world, let's remember that.

Acknowledgments

This work is partly supported by SOCIAM: The Theory and Practice of Social Machines, funded by the UK Engineering and Physical Sciences Research Council (EPSRC) under grant number EP/J017728/2.