Primer on Blockchains and Smart Contracts

February 20, 2019

It's impossible to shop for anything today without coming across a "smart" version of the product you're interested in. Smart TVs, thermostats, security systems, refrigerators and even vacuums are everywhere. The legal industry has not been immune from this terminological fad—smart contracts are one of the foundational projects of the crypto community. However, an explanation of smart contracts is perhaps not as intuitive as other smart products you may have come across. So what are they? And how do they relate to blockchains? This alert is designed as a quick primer on how to understand smart contracts and what to expect from them in the future.

Blockchains

The original idea for smart contracts was proposed by cryptographer Nick Szabo in the mid-90s.[1] The analogy that is commonly used in the literature is of a vending machine. Broadly, a vending machine can be thought of as combining two essential features: a security feature and a contractual feature. The security feature—namely, the big, heavy tower the products sit in to guard against vandalism—is analogous to a blockchain. However, while a giant box and thick glass is usually enough to protect the value contained in a vending machine, it is much harder to find a solution for protecting the value contained in billions of global transactions. Blockchains provide this security by combining two characteristics: decentralization and immutability.

Blockchains can best be understood as (a series of) ledgers which keep track of and verify sets of transactions. Each individual block in a chain contains three important elements: 1) the relevant information about the transactions themselves[2]; 2) a unique "fingerprint" identifying the individual block (i.e. a random hexadecimal string of numbers, or hash, created by inputting the data in the block into an algorithm); and 3) a reference to the previous block's fingerprint, which thereby creates the link in the chain. Because of the way these three elements interact with one another, you cannot alter the data inside a block without altering its fingerprint, nor can you alter a block's fingerprint without altering every subsequent block's fingerprint. Thus, attempting to change so much as a nickel on the ledger requires you to effectively rewrite the entire set of blocks on the chain. Even assuming this was possible in a given instance, the cost of supplying the requisite computing power to accomplish it would be prohibitive, not to mention everyone would be able to see the transaction had been tampered with, immediately devaluing all your hard work. The result is an

ongoing, immutable account of all transactions on a chain packed into each new iteration of the account itself.

A related security feature of blockchains is known as the consensus protocol. Each time a new block is proposed it gets sent to all other computers in the network. Those computers can verify that the information in the proposed block is "true" and once enough of them have done so (i.e. reached consensus), the block is added to the chain.[3] A new block will not be added to the chain unless and until 51% of the other computers agree to the block. This is what it means to say blockchains are decentralized, they don't rely on one authority for their validity, but instead on many other computers all agreeing on the proposed information (all agreeing on a simple math problem). Combining the two essential characteristics of blockchains, immutability and decentralization, gives us a way of keeping track of transactions that does not require trust (which always comes at a cost) and cannot be altered (which is always a benefit).

Smart Contracts

Going back to the original example, the contractual feature of a vending machine gives us a starting point for smart contracts. Vending machines contain hardware that, in effect, have contracts encoded onto it. Put crudely, the encoded contract reads something like, "If and only if you insert $1.00, and then if and only if you type in "B4", then you will receive the product displayed at B4 (e.g., a bottle of water)." The vending machine then also has the hardware to self-execute this agreement should you decide you're thirsty enough to insert $1.00. A smart contract is this same simple idea writ large. Parties agree to contract based on conditional language and that language is turned into a set of ones and zeroes capable of self-executing upon receiving information that a condition has or has not been met

Take a potential real-world example. Suppose Abbi in California decides she wants to buy a shipment of steel from Ben in Japan. Abbi is willing to pay 1M dollars for the steel but needs it before the end of the month. Abbi and Ben can negotiate and enter into a smart contract that, again very crudely, reads something like, "If Ben gets the barge containing the requested steel to the port of Oakland before February 28, then Abbi agrees to pay him 1M." Just as in a real-world negotiation, Ben could also demand, for example, that Abbi be required to put up the 1M (or some smaller amount) on to the ledger as security. The smart contract could be written such that once on the blockchain the security deposit could not and would not move until a condition was or was not met. Ben can therefore be confident he will get paid if and when he delivers the steel because he can see the money on the blockchain—sorry, escrow agents.

Now complicate the situation with a few additional steps. Suppose further that Abbi and Ben want to add another caveat that if the weather in the north Pacific gets bad, Ben gets the benefit of another week to deliver the steel. The contract can then be written such that it can receive weather updates from the ship in real time and, if triggered, update itself to extend the period of the contract. Suppose also that, perhaps

based on the same worries about the weather, Abbi wants to purchase insurance on the shipment. The contract can also be written to allow Abbi to purchase and receive insurance funds upon receipt of information that the shipment has been lost or damaged. Last, supposed still further that Abbi and Ben decide this arrangement is so beneficial they want to agree long term that Ben will send steel to the port of Oakland every month for the next 3 years. The contract can again be written such that it monitors exchange rates and alters payment amounts accordingly. You quickly begin to see the amazing power and potential of smart contracts—they can conceivably carry out any contractual condition imaginable without the need for approval from another party.[4]

However, careful readers also quickly notice a problem here. If the revolutionary features of blockchain and smart contracts hinge on being able to leverage a decentralized, immutable digital infrastructure, what happens when smart contracts have to interact with single parties in the real world to rely on information about the weather in the north Pacific or on exchange rates? The third-party trust problem has re-entered through the back door. This is known as the "oracle problem" and is a *major* problem for future implementation of robust smart contracts. The immutability and trustless efficiency of blockchains are dependent on inputs from humans and other electronic devices (or the internet, for example), all of which are fallible to some extent. Garbage in, garbage out.

Moreover, this is but one of a host of problems that need to be solved. Smart contracts have some very large technical hurdles to jump over before they become part of our everyday transactions. And, of course, there are legal/regulatory issues, too. For example, if a dispute arises between Abbi and Ben concerning their transactions, how will it be resolved? What law governs? What regulatory/judicial bodies have jurisdiction to hear their dispute?

That said, there is no reason to believe all or most of these problems won't be overcome; there's just too much potential in it. If innovations like Uber and Airbnb have unchained idle but otherwise productive assets and put them to good use, we haven't seen anything yet. The idea of a (near) zero transaction cost world is not only conceivable but perhaps even likely at this point; blockchains and smart contracts are a big reason why, and all while minimizing counter party risk and maximizing transparency to boot. Blockchains and smart contracts have the potential to do more than revolutionize the economy (they're already doing that), they may very well change the way we understand and define concepts like "property", "rights" and "contracts" altogether.

---

[1]http://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/

CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/idea.html

[2] In bitcoin for example, it is the identities of each of the parties and how much was sent to whom; more broadly, any type of computer file containing any type of information can be placed on a blockchain.

[3] This is part of the process in bitcoin known as "mining" (it also has various other names in other blockchain projects). There is a growing spectrum of consensus protocols. This wider process is rife with technical issues that need to be solved.

[4] Outside the narrow definition of a contract, smart contract technology can and is being implemented in such differing domains as the airline industry, voting procedures, payroll services, real property and healthcare recordkeeping, investing and, of course, currencies.