

## “Code is Law”

“Code is law” has become a catch phrase in this age of blockchain technology, where transactions of all sorts are being shifted onto blockchain platforms. Some use the term to suggest that code should replace law in many respects in these transactions. Others use it to defend against claims that they have acted wrongly by using technically complex rules to “outsmart” others on a digital platform and obtain outcomes (and sometimes riches) that the large community of users, and developers did not believe could or would occur. This note explores some of the history behind the concept and applications of “code is law,” and implications for the continued role of traditional law as blockchain technology proliferates.

### I. Origins of “code is law”: Lawrence Lessig

Drawing from his 1999 book *Code and Other Laws of Cyberspace*, Lawrence Lessig is credited with coining the phrase “Code Is Law,” which is the title for his 2000 *Harvard Magazine* article<sup>1</sup>. In these writings, Lessig explores issues with the then-nascent Internet, and posits that the absence of government regulation of the Internet does not mean the absence of any regulation. Instead, Lessig asserts that code, written by software engineers, will provide the rules of interaction, and embody value judgments that will set rules for how broader society interacts in cyberspace.

Lessig’s writing came twenty years ago, when less was known about how code and regulation would intersect. However, he was prescient and laid the theoretical foundations for the “code is law” principle that exists today. Lessig’s main argument is that we must collectively understand how code regulates the internet, as it is the basis of our interaction with the digital world. Since code does not employ discretion the way humans do, an automated governance structure will change the nature of law on the Internet. He notes that code can either “embed or displace values from our constitutional tradition,” and that we can ensure that values are not displaced only by understanding the regulator of the internet—code. His article makes three main points:

**1. Law and code must work in tandem to govern the Internet.** The Internet’s governing code protocols (“TCP/IP”) allow data to be shared across networks without identifying the source of the data or any of its substance. This can be both a virtue and a vice: code can uphold freedom of speech by allowing users to remain anonymous. But this anonymity can make it hard to identify and prosecute cybercriminals. However, because code can be altered, it is “net fundamentally regulable.”

**2. “The choice about code and law will be a choice about values.”** Lessig offers an example of a music distribution company spying on the contents of users’ hard drives. This, he explains, is an issue with the architecture the company provides to users. He asserts that only a third-party regulator—the government—could impose a penalty on the company and force them to alter its architecture. Thus, values—through formal law—must have a place in the governance of the Internet, as code itself must be governed.

**3. Code regulates, but people write code.** Lessig argues that we are preoccupied with nonregulation and that we view the concept of regulation as binary; it is either present or absent. According to Lessig, minimal government intervention in cyberspace will not mean less regulation. Absent government regulation, the interests of coders—which may not prioritize shared values, such as privacy—will reign. That

---

<sup>1</sup> <https://www.harvardmagazine.com/2000/01/code-is-law-html>

is, “regulation” will always exist, whether led by coders or by the government. Thus, if the digital world remains “fundamentally skeptical about self-government,” and leaves regulations to coders only, we cannot ensure that our values will be imbedded in the code that undergirds the Internet.

## II. Different views of “code is law”

### a. To what extent can code replace traditional law?

While traditional law sets rules for behavior and retroactively punishes non-compliance, code can “determine what people can or cannot do in the first place.”<sup>2</sup> This theoretically eliminates the need for a traditional legal system to guide behavior, because there is, in theory, nothing to dispute if it cannot happen to begin with. For example, in a blockchain platform smart contract, no one party has control over the operations of the transaction. Every stipulation is structured as an if/then/else statement, meaning that every aspect of the transaction is predicated on the fulfillment of the prior step. There are thus no contractual disputes, as each phase of the contract must be fulfilled to move to the next.

Under a most expansive view of “code is law,” if the code of a smart contract permits something, then it is “legal.” This theory holds that code shall prevail, whether or not it conflicts with anything else. Those who accept his literal meaning of “code is law” contend that, even in the event of a bug or glitch in the code, that same code still governs. Since algorithmic law is unambiguous, they argue that it reduces the subjectivity inherent in the traditional legal and judicial systems. In a code-based system, there is no “flexibility and ambiguity of natural language.”<sup>3</sup> In turn, there is guaranteed consistency in the application of code.

The automation of rules governing transactions and dispute resolution through code, rather than traditional legal avenues, has been cited as viable in some areas, such as in credit card disputes and security deposit agreements. For the former, if a customer were to dispute a charge that was refunded, the case could be closed automatically<sup>4</sup>. Similarly, code operating as law could be used to protect tenants in security deposit disputes. If a landlord fails to explain to a tenant why their security deposit was not refunded, or stopped replying to the tenant all together, there would be an automated initial judgment favoring the tenant<sup>5</sup>. If the landlord had already paid the tenant, a process could verify the relevant financial transaction and automatically close the case. Similar processes could be explored to automate aspects of divorce proceedings, immigration cases, and traffic violation appeals<sup>6</sup>.

But this view has at least one major shortcoming: other than in the simplest of contracts, code cannot necessarily account for every eventuality that might occur in the performance of a contract.<sup>7</sup> Draw downs on letters of credit, for example, which are supposed to be automatic on presentation, are occasionally enjoined for nonperformance, fraud or on other grounds. It is impossible to anticipate, let alone reduce to code, all the scenarios that might unfold. Contracts would have to be hundreds or thousands of pages long to address the

---

<sup>2</sup> <https://journals.openedition.org/factsreports/4518#tocto1n2>

<sup>3</sup> *Ibid.*, 2.

<sup>4</sup> <https://future.a16z.com/law-as-code/>

<sup>5</sup> *Ibid.*, 4.

<sup>6</sup> *Ibid.*, 4.

<sup>7</sup> This presupposes that the contract is also one that could be automated. Many contracts, even simple ones, appear more difficult to completely automate – for example, a contract to paint a house, where payment is due upon completion of the work to the satisfaction of the homeowner.

nuances of every possible scenario. Instead, in contracting we live with deliberate ambiguity,<sup>8</sup> leaving clues for a subsequent reviewing court about “the intentions of the parties.” Concepts and legal terms such as timely delivery, “Class A” quality, no material adverse change, substantial performance, mistake, etc. are usually deliberately left undefined. Not susceptible to definitive explanations in language, they cannot possibly be reduced to code.

Further, code can often be written in a way that leads to consequence that neither the software developers nor the vast majority of community of platform users intended. There can be a sense of fundamental unfairness when an unintended consequence results in huge losses to many, through exploitation by an individual who identifies an unanticipated opportunity in code. Unsurprisingly, almost no lawyers support this viewpoint, as “equities and circumstances are ignored” in this perspective<sup>9</sup>. Whereas software developers see this approach as the fairest, and believe it’s the only way to ensure objectivity in blockchain governance, lawyers see no valid grounds for assessing contracts without context.

### **Example: Ethereum**

A very strong view of “code is law” collided with notions of fundamental fairness in the 2016 Ethereum hard fork. On April 30, 2016, the first digital decentralized autonomous organization was launched, known as “The DAO.”<sup>10</sup> It was built on the Ethereum blockchain with no conventional management structure or board of directors and raised over \$150 million of Ether as of May 21, 2016<sup>11</sup> from more than 11,000 investors<sup>12</sup>. A few weeks later, an Ethereum developer on GitHub, the online repository of The DAO’s open-source code, identified a vulnerability in the code called “race-to-empty.” Essentially, one could technically withdraw multiple times from his DAO account more than his balance in the account because the code would not check the balance in certain recursive withdrawal situations, e.g., when a user is creating a child DAO under The DAO’s umbrella. In this sense, people could “race to empty” the entire treasury of The DAO with a minimal deposit. The DAO was subjected to an attack exploiting a combination of vulnerabilities, including this “race to empty” vulnerability that resulted in the transfer of 3.6 million Ether—around a third of the 11.5 million Ether that had been committed to The DAO—valued at the time at around \$50 million.<sup>13</sup>

The Ethereum community hotly debated whether the attacker’s transactions were legitimate, and eventually decided to undo every transaction involved in the attack by updating the Ethereum code and erasing the problematic transactions from the Ethereum ledger, which lead to the first hard fork of the Ethereum network. The purported attacker alleged that a “hard fork would amount to seizure of my legitimate and rightful ether, claimed legally through the terms of a smart contract.”<sup>14</sup> Those who agreed with the attacker’s view refused to install the software update and continued to use the original unforked Ethereum blockchain,

---

<sup>8</sup> <https://news.mit.edu/2012/ambiguity-in-language-0119#:~:text=Now%2C%20a%20group%20of%20MIT,with%20the%20help%20of%20context.>

<sup>9</sup> <https://abovethelaw.com/2019/08/code-is-law-should-software-developers-protect-our-freedoms/#:~:text=Digital%20Logic's%20CEO%20explains%20how,it%2C%20an%20action%20is%20allowed.>

<sup>10</sup> The capital letter “T” is to distinguish it from later decentralized organizations (DAOs). See [https://en.wikipedia.org/wiki/The\\_DAO\\_\(organization\)](https://en.wikipedia.org/wiki/The_DAO_(organization))

<sup>11</sup> [https://www.nytimes.com/2016/05/22/business/dealbook/crypto-ether-bitcoin-currency.html?\\_r=1](https://www.nytimes.com/2016/05/22/business/dealbook/crypto-ether-bitcoin-currency.html?_r=1)

<sup>12</sup> <https://www.americanbanker.com/bank-technology/the-dao-might-be-groundbreaking-but-is-it-legal-1081084-1.html?zkPrintable=1&nopagination=1>

<sup>13</sup> <https://www.nytimes.com/2016/06/18/business/dealbook/hacker-may-have-removed-more-than-50-million-from-experimental-cybercurrency-project.html>

<sup>14</sup> <https://pastebin.com/CcGUBgDG>

now called Ethereum Classic.<sup>15</sup> The attacker’s transactions are now recorded in the Ethereum Classic ledger as part of the legitimate transaction history<sup>16</sup>. Vitalik Buterin, the co-founder of the Ethereum blockchain, took a different view: “Some Bitcoin users see the hard fork as in some ways violating their most fundamental values. I personally think these fundamental values, pushed to such extremes, are silly.”<sup>17</sup>

## **b. Code leaves a role for traditional law and broader community oversight**

Lessig would endorse the understanding of “code is law” applied by those who implemented the Ethereum fix – that code should have fundamental principles embedded in it. Only by developing code that is in line with our traditional legal standard can we reconcile code and law. Code that governs a transaction between two parties effectuates an agreement to perform between those parties, and therefore could be subject to normal rules of contractual interpretation, and in appropriate cases opportunity for remedial measures. As coders potentially replace the role of lawyers and adjudicators in some contexts, it is important that the public have a consistent say in how the governing code is constructed. Much like democratic legislators, some blockchain communities can participate in code changes or amendments, to provide some of the value-based input and oversight that Lessig wrote about. They can vote for implementing new “laws” embodied in code, just like the Ethereum community voted for the hard fork after The DAO attack<sup>18</sup>. And participants who have sufficient choices can vote with their feet, by steering clear of blockchain platforms that contain undesirable or unduly inflexible code-based rules.

While this definition of “code is law” balances automation while preserving fundamental ideals, it cannot be fully realized yet due to an information problem. In code, information is either syntactic or semantic. Syntactic information refers to a digital character and its relation to other characters, whereas semantic information is the meaning of a certain set of characters given its context<sup>19</sup>. Semantic information is much more difficult—if not impossible—to capture in code, as it is the very point of differentiation between human and artificial intelligence. Thus, since basically all law is semantic information, translating traditional law into code poses a significant challenge, even with machine learning.

## **III. Code is law “light”**

As Joshua Browder expresses in “Law as Code: A Legal System Shaped By Software,” a technology-driven approach to the law could help the legal system function in a fairer way, at least in some simple scenarios. Smart contracts, for example are trustless, efficient, and cut out the middleman. A software-driven legal system would, in theory, provide a more objective and efficient approach to some contracts and other routine legal procedures. Various parts of immigration and divorce law, for example, could be automated where those fields follow routine procedures since some of these procedures are Boolean—they require data collection and a series of questions that elicit one of two possible responses.

But, as noted above, one can quickly envision scenarios where disputes would arise out of ambiguities or unforeseen situations which would require the application of traditional law. Contract disputes are typically

---

<sup>15</sup> <https://paxful.com/blog/what-are-ethereum-hard-forks/#:~:text=Ethereum%20Classic%20is%20the%20very,being%20drained%20from%20the%20fund.>

<sup>16</sup> <https://medium.com/coinmonks/ethereum-classic-code-is-law-f07f183d5f42>

<sup>17</sup> <https://www.bloomberg.com/features/2017-the-ether-thief/>

<sup>18</sup> <https://web.archive.org/web/20180609110235/https://www.sec.gov/litigation/investreport/34-81207.pdf>.

<sup>19</sup> <https://www.coindesk.com/markets/2016/08/27/code-is-law-not-quite-yet/>

based on context and semantic information, and therefore cannot be translated into code without material compromise. It is impossible to code for all eventualities and decision trees that may be implicated by contract terms. These issues require legal professionals to leverage their knowledge of precedent, prior casework, and legal theory to develop a winning strategy, which artificial intelligence and computer code are not yet capable of. Thus, although “code is law” may be defined as traditional law embedded in code, this approach will not provide a solution to most legal problems.

#### **IV. The Code is law debate enters the courtroom.**

In December 2021, a DeFi protocol, Indexed Finance, was exploited by a Canadian teenager, who leveraged an approximately \$159 million flash loan to allegedly distort the price of assets on Indexed Finance and made approximately \$15.8 million profit<sup>20</sup>. When the co-founder of the DeFi protocol sued the exploiter in a class action in Ontario,<sup>21</sup> the exploiter claimed “code is law,” arguing that the DeFi ecosystem is regulated based on code’s permissibility.<sup>22</sup> In a series of tweets, the purported exploiter argued that the transactions were simply “a few preprogrammed price updates.”<sup>23</sup> He tweeted: “The people who I traded against and won money from read the same contract I did and were willing to deploy their capital on it. At no point in this process did I do anything that could not have been done by anyone else.”<sup>24</sup>

The plaintiff co-founder of Indexed Finance expressed a contrary view (more in line with Ethereum’s co-founder, in relation to the 2016 Ethereum hack):

I consider [code is law] a fringe and unworkable view of how DeFi actually needs to work if it’s going to be more widely adopted .... A lot of the “code is law” appreciators seem to think that DeFi stands entirely outside of the framework of law, as opposed to simply being a manner of disintermediating financial institutions[.] Just because all of this happens through a blockchain doesn’t mean that suddenly hundreds of years of legal process magically stop applying.<sup>25</sup>

To date, the exploiter has not appeared before the Canadian court. When issuing a warrant for the arrest, the Canadian Judge stated:

Refusing to participate does not indicate a good faith belief in the justice of one’s cause. If [the exploiter] wants to assert that the code speaks or the code is law, he has to participate in the lawful process pending the outcome of the debate.

...

He is a young man whom, I fear, is caught between the law and a set of rules that he asserts operates independently. The only way he can show that he is or ought to be held to be correct is by participating and making the case that he asserts<sup>26</sup>.

In a separate January 17, 2022 order, the Judge further identified the interesting issues that the case will present, if the attacker chooses to participate:

---

<sup>20</sup> [https://drive.google.com/drive/folders/1B5Zy-xt\\_KrQGKQ1ZhukBPXB-sIXfmTBW](https://drive.google.com/drive/folders/1B5Zy-xt_KrQGKQ1ZhukBPXB-sIXfmTBW).

<sup>21</sup> In December 2021, the Canadian court joined this action with another case also pursuing the attacker (<https://www.canlii.org/en/on/onsc/doc/2021/2021onsc8473/2021onsc8473.html>).

<sup>22</sup> <https://coingeek.com/canadian-teen-faces-arrest-warrant-over-defi-hack-in-code-is-law-case/>.

<sup>23</sup> <https://twitter.com/ZetaZeroes/status/1451065309213962247>.

<sup>24</sup> <https://twitter.com/ZetaZeroes/status/1451065309213962247>.

<sup>25</sup> <https://blockworks.co/canadian-15-8m-defi-theft-case-could-upend-code-is-law-convention/>.

<sup>26</sup> <https://www.canlii.org/en/on/onsc/doc/2021/2021onsc8473/2021onsc8473.html#document>

There is a theory in some cryptocurrency academic thought, that because blockchain technology is based on publicly available or “open source” programming code and is based on a laissez-faire contract theory, that “the code is law”. That means, that if one is able to trade with a blockchain participant within the parameters of the programming code or the notional contract among the voluntary participants, the result is lawful whatever it may be.

The theory postulates that voluntary participants accept and are bound by the results of the use of the technology. That means that if a clever persona can devise a way to exploit a loophole or weakness in the code to induce the holder to enter into an unexpected and unfavourable transaction, more power to him or her. The code is public and the users are deemed to take the risk of placing their cryptocurrency assets in a repository with a program that functions as it does with whatever vulnerabilities it may have.

...

Whether the Ontario common law supports this legal theory might well be in issue in this action – if [the defendant] participates<sup>27</sup>.

To date, the public docket does not show further meaningful activity in this case.

## V. The future

As blockchain technology continues to proliferate and expand to new use cases, opportunities for code to become a governing structure and provide rules for certain transactions will increase. Some will argue for the notion that code can replace traditional law across many interactions that move to blockchain technology. However, it is unlikely that code will replace law in any large sense. Governments keen on regulating will not entirely allow it, and the variety of disputes that can arise—based on ambiguities in language, unforeseeable outcomes, and flaws in coding—will create situations where aggrieved parties will seek (and need) the help of the legal system. How courts and lawmakers draw lines remains to be seen, particularly in cases where smart individuals claim to have followed the “rules” set out by the code.

\*\*\*

To view more memoranda, please visit [www.quinnemanuel.com/the-firm/publications/](http://www.quinnemanuel.com/the-firm/publications/)  
To update information or unsubscribe, please email [updates@quinnemanuel.com](mailto:updates@quinnemanuel.com)

---

<sup>27</sup> <https://www.canlii.org/en/on/onsc/doc/2021/2021onsc369/2021onsc369.html>