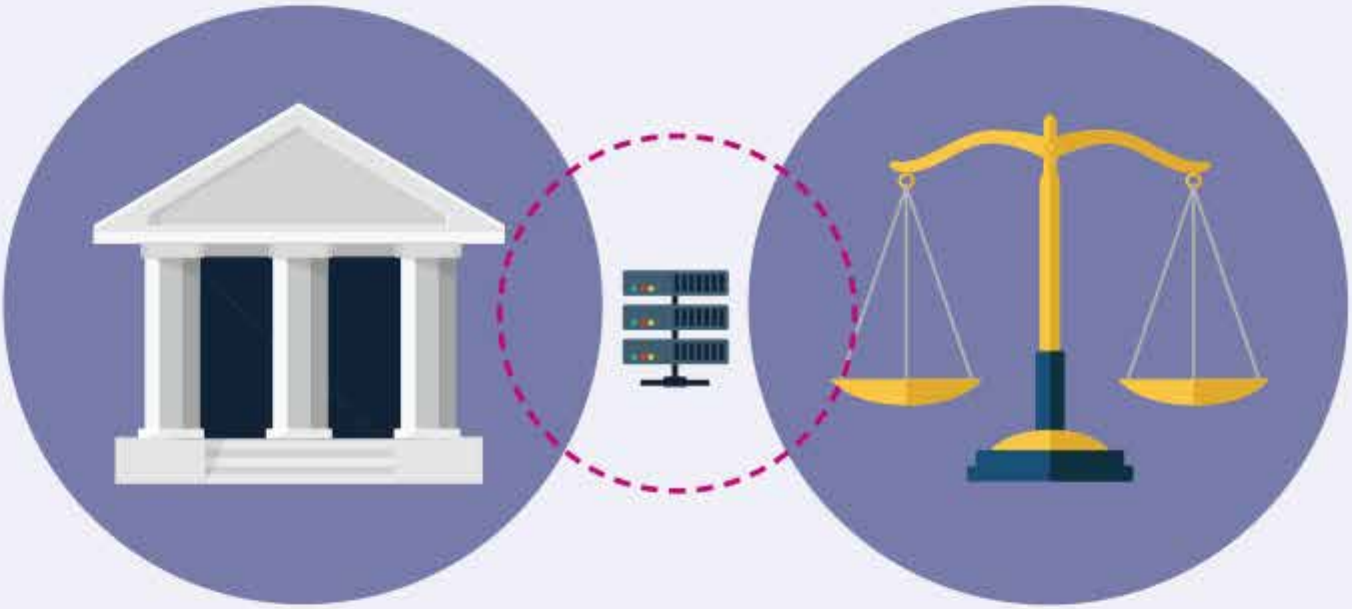
CHAPTER 3

Governance and Regulation



Both the legal and the digital spheres are governed by rules, but the nature of these rules is different. In a digital environment, both laws (legal code) and software/hardware (technical code) regulate activity. The impact of both must be considered in setting out regulations that cover distributed ledger systems.

Author
Vili Lehdonvirta, Oxford Internet Institute, University of Oxford;
Robleh Ali, Manager – Digital Currencies, Bank of England

Chapter 3: Governance and Regulation

Introduction

This chapter deals with rules and rulemaking in distributed ledger systems. We will distinguish between **legal code** (rules consisting of legal obligations) and **technical code** (software and protocols). We will also distinguish between **governance** (rule-making by the owners or participants of a system with the purpose of safeguarding their private interests) and **regulation** (rule-making by an outside authority tasked with representing the interests of the public).

Legal code vs technical code: Two types of rules

The financial system is both a set of legal obligations between institutions and a set of digital records of these obligations. Both the legal and the digital spheres are governed by rules, but the nature of these rules is different. In a seminal text on the subject[1], Lawrence Lessig of Harvard University addressed how these legal and digital rules interact to govern activity. Lessig argued that in a digital environment both laws (legal code) and software/hardware (computer code) regulate activity, and that the impact of both needs to be considered when constructing a theory of regulation. In this chapter we refer to technical code rather than computer code. This definition covers both software and protocols, as distributed ledgers rely on both to function.

One fundamental difference between legal code and technical code is the mechanism by which each influences activity. Legal code is 'extrinsic': the rules can be broken, but consequences flow from that breach to ensure compliance. Technical code, in contrast, is 'intrinsic': if its rules are broken then an error is returned and no activity occurs, so compliance is ensured through the operation of the code itself. Another characteristic of software is that a machine will rigidly follow the rules even where that compliance produces unforeseen or undesirable outcomes. This leads to some striking differences in the operation of distributed ledger systems compared with the current financial system.

1. Current financial system: ruling via legal code

The modern financial system is already largely digital and heavily reliant on technical code. This technical code governs the creation and amendment of the digital records of the legal obligations between institutions. Financial regulation is aimed at the effects these legal obligations produce: for example, whether a bank has sufficient capital or liquidity. The financial system is already governed by this combination of technical code and legal code, but financial governance and regulation has traditionally focused on the latter.

Enforcement of the public element of the legal code falls to a specialised group of financial regulators charged with ensuring compliance by participants in the system. Participants must provide the information that their regulator needs to assess whether they are in compliance with the system's rules. If an institution is not in compliance then the regulator can take action to bring them back into line. This is not to say technical code has no influence on the existing regulatory process — all the information provided to the regulators is digital, and the product of technical code — but governance and regulatory aims are pursued by producing legal code rather than by changing the technical code.

2. Distributed ledger systems: ruling via technical code

Distributed ledger systems such as Bitcoin have shown that they can function without legal rules. Instead, the rules that each participant must follow are defined and enforced only by technical code. Each participant in the network runs the same or compatible software that defines what kinds of transactions are permissible. For example, the Bitcoin software allows participants to spend only balances that they can prove they own with cryptographic keys. The Bitcoin software also regulates how new currency is issued, and places an absolute cap on the size of the money pool. There are no bylaws or other legal documents stating these rules, and no humans to enforce them — distributed ledger systems are solely governed by their own technical code.

To prevent participants from modifying their copy of the code to issue transactions that are against the rules, each transaction needs to be verified before it enters the ledger. In an 'unpermissioned' distributed ledger system like Bitcoin, verifiers (known as miners) are chosen by lottery. The system seeks to assure their integrity through a system of economic incentives, in a process governed by the software. In a 'permissioned' distributed ledger system, verifiers are appointed by the system's proprietor, and their integrity is assured through conventional means, such as a legal contract.

In summary, distributed ledger systems differ from the conventional financial system in that they are ruled by technical code rather than legal code. One advantage of this is that compliance costs are low: participants need only use a compliant software package to issue transactions. It might seem that enforcement costs are lower, too, but this is not necessarily the case because the mining system that is used to verify transactions in all of the most popular distributed ledger systems consumes significant computational resources. That cost must eventually be borne by the system's users.

Governance vs regulation: Two types of rule-making

Because the current financial system and distributed ledgers are primarily governed by different types of rules, we must therefore ask the question: who makes the rules?

1. Current financial system: a mesh of private and public rule-making

There are many places where legal code is being produced in the current financial system, but these can be broadly divided into two categories: private rule-making (governance) and public rule-making (regulation). An example of private rule-making is the Visa Core Rules promulgated by the financial services company Visa Inc. to govern the actions of all the participants in the Visa system. Such private rule-making is done by proprietors of private financial networks like Visa, as well as by private associations of financial institutions wishing to coordinate their activities to one another's benefit. An example of public rule-making is the statutory oversight of Visa Europe's payment system by the Bank of England.

The design of the public legal code in the current financial system is the province of policymakers who have to consider the effect of regulations on the different institutions of the financial system (a 'microprudential' approach) as well as the impact on the system as a whole (a 'macroprudential' approach). As the financial system is global, international bodies such as the Basel Committee on Banking Supervision convene policymakers from around the world to reach voluntary

accords that can then be translated into legislation in a specific jurisdiction.

2. Distributed ledger systems: ad hoc private rule-making

Unpermissioned distributed ledger systems are sometimes thought to exist independently of human rule-making, and governed only by mathematical algorithms. This is a misconception. Just like legal code, technical code needs to be produced and maintained by humans who define the rules that the code embodies. Using Bitcoin as an example, the initial version of the software was published by Satoshi Nakamoto (a pseudonym). In 2010, Nakamoto handed control of the project to Gavin Andresen, an Australian-born programmer living in the United States. Like any software, Bitcoin needs to be regularly updated to address bugs, security issues, and changes in the operating environment. Such an update can in principle change any aspect of the software, including accounting and ownership rules. Who gets to write the software and how that process is governed is therefore critically important to all participants in a distributed ledger system.

In the case of Bitcoin, the software is governed by an ad hoc process involving a handful of informal institutions and power holders. Figure 1 shows who has written most of the current Bitcoin code. The software is open source and anyone can suggest changes to it, but technical authority to admit changes to the official version of the software is held by a team of five core developers appointed by Andresen. The core developers' power is constrained by an informal self-imposed charter, which states that significant changes to the rules require broad consensus from the community. Any update to the software must furthermore be installed by a majority of the miners (as measured by the computer processing power they contribute) for the changes to become effective. A handful of individuals who manage so-called mining pools are therefore very influential in determining whether or not miners ratify a software update in this way.

This governance process worked well when the changes to the code were uncontroversial bug fixes, but it has started to show signs of breaking down recently, because some decisions require choosing which stakeholders' interests to prioritise over others'. Andresen and others have stated that the process needs to become more formal. The community is debating what such a formal governance system should look like, but this is complicated by the fact that

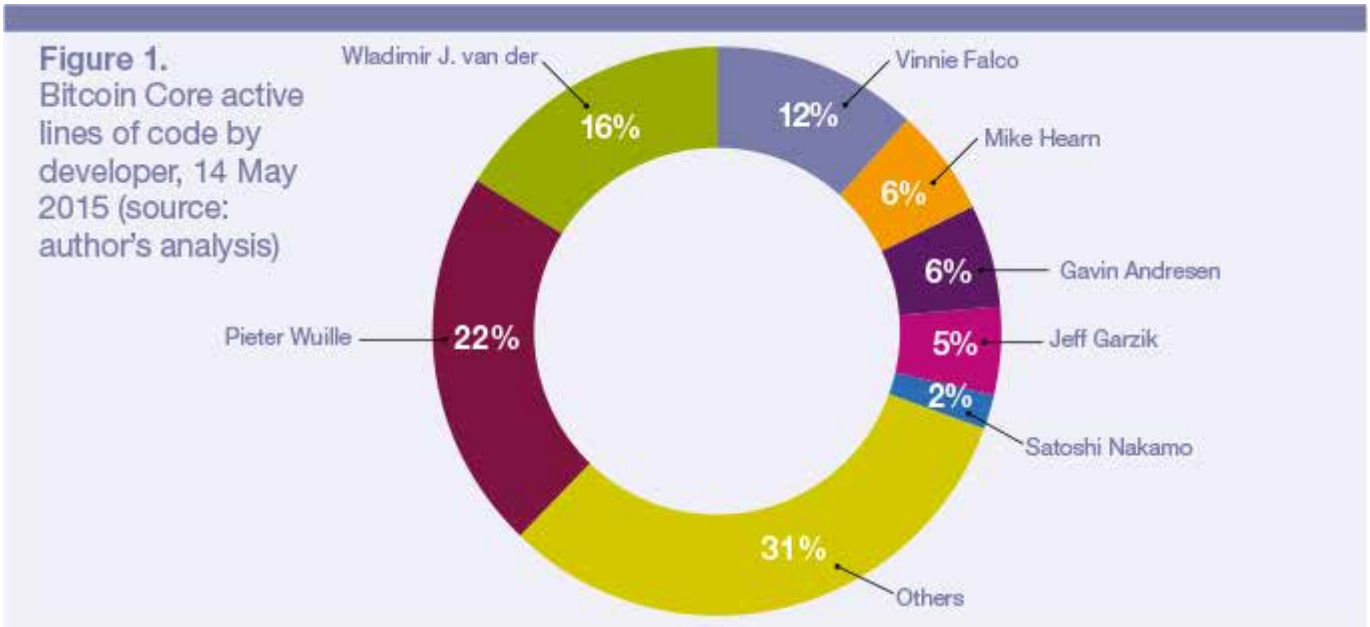


Figure 1. Bitcoin Core active lines of code by developer, 14 May 2015 (source: author's analysis)

Bitcoin was founded on an ethos of anti-institutionalism. This is an interesting conundrum, as it demonstrates the worth of legal code and shows that technical code alone does not produce an optimal outcome.

In permissioned distributed ledger systems, governance of the software is made simpler by the fact that there is usually a proprietor with clear legal and technical authority over the code. It is up to the proprietor to determine how the code is modified, and up to the users (often customers of the service) to decide whether they are comfortable with having the proprietor exercise authority over the software. Service level contracts and other conventional means can be used to establish responsibilities and enforce them. Permissioned distributed ledger systems are in this respect not very different from conventional private financial networks like Visa or software-as-a-service (SaaS) systems.

How should we regulate distributed ledger systems?

Governance in a distributed ledger system as described above is concerned with the system's stakeholders' interests, but there may also be broader social interests involved in how a distributed ledger functions. For example, regulators may wish to collect taxes, prosecute crimes, and limit the use of a distributed ledger for criminal purposes. If a system is adopted to the extent that it starts to have potential knock-on effects elsewhere in society, regulators may also wish to ensure that the system is resilient against systemic risks and market failure. This regulation can be applied through legal code or technical code.

1. Regulating distributed ledgers via legal code

Regulating a permissioned distributed ledger system is simply a matter of imposing legal obligations on its proprietor. Regulating an unpermissioned system like Bitcoin via legal code is more complicated, as there is no single legal entity in control of the system. It would be difficult to regulate what software people are allowed to install on their computers. Attempts to regulate Bitcoin via legal code have instead focused on regulating the businesses that deal with Bitcoin, such as exchanges and wallet providers. These businesses can be regulated in their own right (eg to prevent a wallet provider from disappearing with customers' money) or as a means to indirectly regulate what the ledger is used for (eg ensuring compliance with anti-money laundering regulations).

A well-known example of regulating Bitcoin via legal code is the BitLicense, issued by the New York State Department of Financial Services to businesses offering digital currency services to New York residents[2]. The deadline for businesses to obtain the license was 8 August 2015, and unlicensed service providers can be penalised.

2. Regulating distributed ledgers via technical code

The technical code for distributed ledger systems like Bitcoin is currently produced by private actors in an ad hoc process. But technical code, comprising software and protocols, can also emerge from the public sector. For example, TCP/IP and some other core internet protocols were the result of government-funded research projects and are now maintained under the auspices of the Internet Society, an international non-profit organisation with an open membership structure based on geographic location and special interests. Other parts of internet infrastructure are maintained by international multi-stakeholder processes, and some parts remain under the oversight of US public regulators.

While this patchwork is far from a perfect solution, it points to the possibility of public involvement and democratic representation in the production of technical code — public regulation via technical code as opposed to legal code.

Table 1
Examples of privately and publicly produced legal code and computer code

	Legal code	Protocol
Privately produced	Visa Core Rules Faster Payment Service Rules	Financial Information eXchange (FIX) protocol Bitcoin
Publicly produced	European Market Infrastructure Regulation BitLicense	Internet (TCP/IP) World Wide Web (HTTP)

Applied to distributed ledger systems, this could mean anything from instituting formal multi-stakeholder processes for maintaining the technical code, to developing public standards for the code. If this allowed governments or the public directly to attain legitimate regulatory goals by influencing the rules built into the computer code, it could lessen the need for a body of new legal code to regulate these systems.

Alternatively, the public sector could develop a permissioned system that allows public regulatory influence to be exerted through a combination of legal and technical code, rather than exclusively through legal code as at present. Some of the core internet technologies have shown that it is possible for governments to successfully catalyse the creation of technical code that has become foundational to private sector activity.

Conclusions

In contrast to conventional private financial networks like Visa, unpermissioned distributed ledger systems like Bitcoin lack a central legal entity with formal responsibility over the system. Instead, they are governed by ad hoc processes, usually centring on a handful of software developers who produce the system's software code. If these systems are to grow in value and influence, they will most likely need to develop more robust internal governance processes. The lack of a central legal entity also makes it more challenging for public regulators to regulate distributed ledger systems via legal code. Governments should therefore also consider ways of regulating distributed ledger systems by influencing the technical code that defines their rules. In finding the right blend, the government should consider the strengths and weaknesses of both technical code and legal code, recognising that the two interact and should be designed accordingly.

The emergence of Bitcoin and distributed ledger systems has brought the issue of technical code to the fore in the context of the current financial system as well. Distributed ledgers show that financial systems can be governed and regulated with technical code as well as legal code. Policymakers should recognise the influence of technical code on the financial system and consider how such influence could be made part of the regulatory system, with potential benefits such as lower compliance costs.