LAWTECH UK

UK Jurisdiction Taskforce's Control Panel

Report

on Control of Digital Assets

Ministry of Justice

LAWTECH UK PANEL

The UKJT's Control Panel Report on Control of Digital Assets

March 2026

Contents

Foreword by Sir Geoffrey Vos, Master of the Rolls

I am delighted to introduce the UK Jurisdiction Taskforce's Report on Control of Digital Assets. The Report has been prepared by an expert panel of lawyers and technological experts led by Sir Antony Zacaroli.

The other members of the Control Panel are Lawrence Akka KC, David Quest KC, Peter Hunn, Richard Brown, Sarah Green, Mez Raja, Elli Androulaki, Ferdish Snagg and Gavin Thomas. Judy Fu led a drafting team including Richard Brown, Elli Androulaki and Gavin Thomas.

I am extremely grateful to Sir Antony and all the members of the Control Panel for their untiring dedication and insight.

In early 2024, the Lord Chancellor requested the UKJT to prepare a report on the Control of Digital Assets under the law of England and Wales. That request followed up on a suggestion made by the Law Commission in its final report on Digital Assets published on 27 June 2023.

I believe that the Report achieves its purpose of explaining with conspicuous clarity how the legal concept of control applies in fact and in practice to digital assets under English law.

Sir Geoffrey Vos

Master of the Rolls and Head of Civil Justice in England and Wales

GLOSSARY

The Panel offers the below glossary of frequently used terms which may be less familiar to a non-technical audience. However, whilst such terms are defined for ease of reference, the Panel's experience is that many terms in the digital assets industry are often used interchangeably or inconsistently in practice. In this Paper the Panel does not impose a level of precision that does not exist in real-world usage, preferring to introduce and explain the underlying concepts. The reality is that what any individual means when using one of these terms may differ.

Litigants may therefore need to be asked explicitly what they mean by a particular term in the context of a dispute, to avoid misunderstanding.

BINANCE	A well-known crypto exchange
BITCOIN	A well-known cryptocurrency
BLOCKCHAIN	A shared online ledger in which transactions are recorded in time-stamped blocks that are 'chained' together, making it hard to change past records and easy to track ownership of digital assets. Different blockchains represent and transfer value in different ways
COINBASE	A well-known crypto exchange and custody provider
CRYPTO ASSETS OR CRYPTO CURRENCY	A digital asset of a blockchain system, typically used as a medium of exchange
CRYPTO EXCHANGE	A platform enabling users to buy, sell, and hold crypto assets
DAO	Decentralised Autonomous Organisations, a governance framework through which participants can make collective decisions through (for example) a protocol, treasury, or application. Degrees of decentralisation vary between DAOs

DIGITAL SIGNATURE	A cryptographic tool that proves the signatory control of a private key and authorises a specific transaction without revealing the key itself
ENCRYPTION	A way of scrambling information so that only someone with the necessary code or key can access it. This protects information from unauthorised users
ETHER or ETHEREUM	A well-known cryptocurrency
LEDGER	A provider and brand of hardware wallets
NFT	Non-fungible token. This is a unique digital token on a blockchain that functions as a unique identifier, which proves authenticity of a specific digital item (such as a piece of digital art or a collectible) so that it can be bought and sold like a one-of-a-kind asset
PERMISSIONED SYSTEM	A blockchain or distributed ledger where only approved participants can join, access data, or validate transactions
PERMISSIONLESS SYSTEM	A blockchain or platform where anyone can participate (subject to protocol rules) by joining the network, holding a wallet, or helping verify transactions without needing prior approval To be clear, the term "public" in this Paper refers to blockchains that are open to everyone, to participate either in transaction validation or to get a copy of the ledger. This is to be contrasted with "private" blockchains, where access is restricted to specific, properly authorised entities. With that said, the term "public" is often used to refer to "permissionless" blockchain systems, as permissionless blockchains are all public. Permissioned systems, on the other hand, can be private or public, e.g., by allowing ledger access to a wider audience

PRIVATE KEY	Secret cryptographic key material that enables a user (or system) to generate valid digital signatures to authorise transactions
PROTOCOL	In this context, a core set of rules and standards that governs how a blockchain network functions – e.g., defining how transactions are validated, blocks are added to the blockchain, and data is shared securely between participants
QUANTUM COMPUTING	In this context, a class of computing that may, in future, weaken existing key encryption methods
SEED PHRASE	A 12 to 24 mnemonic word sequence generated by a wallet, which acts as a back-up and allows access to digital wallets to be recovered. Anyone with the seed phrase can typically recreate a wallet's signing keys
SMART CONTRACTS	In this context, a computer code deployed to a blockchain which makes changes on-chain according to defined rules. Smart contracts can automatically perform actions (e.g., releasing payment) when pre-set conditions are met, allowing digital assets transactions to be executed without manual intervention Not to be confused with smart legal contracts (a type of smart contract), a legally binding agreement in which some or all contractual obligations may be defined in and/or performed automatically by a computer program
STABLECOIN	A type of cryptocurrency which is designed to maintain relatively stable value, usually by being pegged to fiat currency (e.g., US Dollars) or to the value of commodities
TREZOR	A provider and brand of hardware wallets
WALLET	A software (such as an app on a mobile phone) or hardware device which holds private and public keys, letting a user access, send, and receive digital

assets. Wallets do not store the actual digital asset (which exists only on the blockchain), but manage the keys required to prove ownership and authorise transactions securely

INTRODUCTION

1. In its 2023 Final Report on Digital Assets (Law Com No. 412), the Law Commission of England and Wales recommended the establishment of a panel of experts, legal practitioners, academics and judges to provide non-binding guidance on the evolving factual issues relating to control of third category things such as digital objects.

2. It recommended as follows:[1]

> "*We conclude that common law jurisprudence will be enhanced and made easier to understand for market participants by focusing on better descriptions and real world examples of factual control. The technical expert group could describe the different factual ways in which control is used in the market today (and, crucially, update these descriptions as the market and technology evolves). It could undertake this exercise on the specific understanding that its output will be helpful for the judiciary when considering legal issues relating to its control. It would therefore need to balance technical accuracy with accessibility to legal professionals.*"

3. And further:[2]

> "*The purpose of the technical expert group considering these types of control is to provide the legal market and the judiciary with a common toolkit and factual reference point from which to develop clear and consistent legal analysis. Rather than relying on inaccurate analogies (such as an analogy with a physical object in a warehouse, the door which has multiple keys; or a traditional bank account; or a bag of gold), market participants and courts will benefit from an analysis of the actual technology in question. This will force claimants and defendants more accurately to disclose and present their case, which should lead to clearer, more logical and more consistent applications of legal rules and reasoning over time. This is fundamental to our conclusion that the common law is the most appropriate mechanism for legal development with respect to the nuanced features of digital objects. A principal goal of the technical expert group could be to maintain a repository of material that considers different factual scenarios and ensures that technological analysis is in line with existing market practice and proposed future developments.*"

4. In accordance with the Law Commission's recommendation, in April 2024, the Lord Chancellor invited the UK Jurisdiction Taskforce ("**UKJT**") to constitute a technical panel (the "**Panel**") to produce a report explaining how the legal concept of control applies to digital assets in fact and in practice. The UKJT formally established the Panel in April 2024, which convened for the first time in December 2024. The Panel's

---

1 Final Report, paragraph 5.27.

2 Final Report, paragraph 5.34.

members are Lord Justice Zacaroli (Chair), Lawrence Akka KC, David Quest KC, Peter Hunn, Richard Brown, Sarah Green, Mez Raja, Elli Androulaki, Ferdish Snagg, and Gavin Thomas. Together with Richard Brown, Elli Androulaki, and Gavin Thomas, Judy Fu of 3 Verulam Buildings served as the lead drafter of this Paper.

5. To supplement its own views, the Panel conducted informal consultations between June-September 2025 with experts from Ankura, Ten, A16z, AlixPartners, and BRG, followed by a targeted consultation in February 2026. The results of that consultation were analysed and considered by the Panel.

CHAPTER I. LEGAL CONTEXT FOR CONTROL

6. In its Final Report on Digital Assets, the Law Commission of England and Wales concluded that certain digital assets do not fit comfortably within either of the traditional legal categories of "things in possession" or "things in action". Nevertheless, under the law of England and Wales, such assets — referred to as "third category" digital assets — still require proprietary protection. The Law Commission recommended statutory confirmation that an object should not be denied recognition as an object of personal property rights merely because it is not easily accommodated within the traditional taxonomy.[3] It decided, however, not to attempt to define the third category in legislation, leaving the boundaries of it instead to be determined through the incremental development of the common law.

7. The factual ability either to exclude or to permit access to a thing is fundamental to a property right. Historically, this ability has been identified by reference to the concept of possession. In its Final Report, however, the Law Commission describes (but, again, does not define) control rather than possession as the factual concept that best captures the ability to:

   a. exclude or permit access to a third category thing; and

   b. put a third category thing to the uses of which it is capable.

8. In the Consultation Paper that preceded the Final Report, the person in control of a third category asset was described as the person who is able sufficiently to:[4]

   a. exclude others from the [digital] object;

   b. put the [digital] object to the uses of which it is capable; and

   c. identify themselves as the person with the abilities specified in (a) and (b) above.

9. The Report concludes that this is not a rigid definition that should be put on a statutory footing, but should instead form the basis of judicial development. It then goes on to

[3] This is now incorporated in the Property (Digital Assets etc) Act 2025, which received Royal Assent on 3 December 2025.

[4] Law Commission Final Report at 5.10.

discuss the legal significance of the concept of control over third category things and makes it clear that there is scope for that category to accommodate a wide variety of different types of asset. The Law Commission did not, however, recommend that third category things should themselves be defined by the concept of control, not least because control is a concept that is complex, composable and multi-faceted, and because different technology, products and services use control in different ways. The Commission therefore recommended that the Government create a panel of industry-specific technical experts, legal practitioners, academics and judges to provide non-binding guidance on the complex and evolving ways in which control can be exercised in relation to third category things.

10. The Law Commission clearly defined control as a factual matter; one that functions as an important constituent element of more complex legal mechanisms or structures. This mirrors the position taken by the UNIDROIT Working Group and the U.S. Uniform Commercial Code (UCC):

> "*The … requirements … contemplate that 'control' assumes a role that is a functional equivalent to that of 'possession' of movables. However, 'possession' in this context is a purely factual matter and not a legal concept. Moreover, because a digital asset is intangible, this functional equivalence to possession involves only the dominion and power over a digital asset but does not involve the physical situs dimension applicable to possession of movables. Whether 'control' … exists is a matter of fact and does not depend on a legal conclusion. However … the presence of control gives rise to legal consequences*."[5]

11. Control, as defined in Principle 6 of the UNIDROIT Principles on Digital Assets and Private Law, consists of:

    a. the exclusive ability to prevent others from obtaining substantially all of the benefit from the digital asset;

    b. the ability to obtain substantially all of the benefit from the digital asset; and

    c. the exclusive ability to transfer these powers to another person.

12. In the U.S., the 2022 amendment to the UCC, in the form of new Article 12, provides a detailed framework for "controllable electronic records" ("**CERs**"), a subset of digital assets. Under this framework, control over a CER enables transfer of ownership; allows

[5] UNIDROIT Working Group, Principles on Digital Assets and Private Law (2023), p 38 para 6.2.

a transferee to take free of third-party claims, if acquired in good faith for value; and serves as a method of perfection and priority in secured transactions. A person has control over a CER if they can avail themselves of substantially all the benefit of the record; have exclusive power to prevent others from doing so and to transfer control; and can be readily identified as having these powers (e.g. via cryptographic key).[6]

Relationship between control and possession

13. In English law, possession refers to a combination of physical control and intention to possess. It is most commonly associated with tangible assets, where the person in possession is presumed to have a proprietary interest enforceable against all but the true owner. Possession is both a factual state and a legal conclusion—its determination depends not only on control, but on broader considerations such as the context of custody, the parties' intentions, and legal policy. Although rooted in physical control, possession can also arise in constructive or legal forms. For instance, a bailee may have possession of goods despite not physically holding them, and an employer retains possession of a laptop used remotely by an employee. These examples illustrate that possession is not reducible to mere physicality; it also reflects a socially and legally recognised relationship to the thing.

14. Bridge notes that possession defies rigid definition: "*a legal conclusion [about possession] cannot be said to follow from a neutral evaluation of fact*."[7] Factors typically considered by Courts include:

    a. physical control over the object;

    b. knowledge of the object's nature and location;

    c. the possessor's intention;

    d. the legal relationship between the possessor and the location of the asset; and

    e. broader legal and policy considerations, including the need to protect security and stability of commercial transactions.

---

6 §12-105.

7 M. Bridge, Personal Property Law, 3rd ed. (Oxford, Oxford University Press, 2002).

15. The Law Commission proposals treat control as a (factual) relationship that a person can have with a third category thing — an analogous concept to possession. Significantly, control recognizes the same legally relevant characteristics of the relationship between person and thing, but does so without the historical baggage that comes with possession; a concept that is now perhaps inextricably linked with corporeal things. Although it is complex, the concept of possession has remained a remarkably flexible tool for the law of England and Wales and the current law would almost certainly not have been able to develop the necessary nuance had possession been defined in legislation. The Report therefore recommended retaining this flexibility for third category things, aiming to ensure that the law asks no more of third category things than it does of things in possession. Where third category things are concerned, the Law Commission determined that factual control is the concept that functions as the most effective analogue to possession in the context of third category things. It will be crucial for Courts to recognise that control over third category things is manifested and mediated by software and can work differently to control over things in possession and things in action. To ignore the fundamental design-features of third category things (particularly crypto-tokens that exist by reference to public, permissionless and distributed crypto-token systems) is to risk the law evolving at odds with the expectations and intentions of market participants. Instead, it is better for the law to accept the reality of modern technology and to adapt to it a concept of control that is functionally equivalent to possession in a way that is responsive to the peculiar features of different technologies.

16. The Law Commission anticipated that the Courts will invoke the concept of control as a matter of default. It made the point in its Report that the Courts will be able to draw on, if necessary, analogous case law in other jurisdictions, and international law reform initiatives, such as the UNIDROIT Working Group's Control Principle, to help them develop the concept of control under the law of England and Wales. Significantly, the UNIDROIT Working Group does not explicitly recommend that Member States adopt a statutory definition of control: it instead frames its Control Principle as a broad guiding principle. Given that the common law has developed the principles of possession (and factual control) over time, in response to market developments and legal challenges, the Commission took the view that it would be able to do so again in the context of digital assets.

17. In prescribing a largely common law approach to accommodating digital assets, the Law Commission recognised that the courts are bound to encounter difficulties in making complex findings of fact where control works differently for different third category things and through different technological implementations. For instance, the Report suggested that the temporal dimensions of crypto-token transfers might have to be carefully considered if control is accurately to apply to, for example, crypto-token systems which may use different methods of establishing a canonical and chronological order of transactional events or state-changes. This might be the case, for example, in relation to the means of effecting a transfer or divestiture of control. Other situations which could give rise to added complexity include the receipt of unexpected (or unknown) crypto-token airdrops, and crypto-token system or network downtime. This shows how digital assets in the third category are likely to give rise to a fundamental challenge that the law will have to acknowledge. That is, technology now facilitates the creation of intangible things of value that are programmable, are rivalrous and can be transferred, stored or traded electronically on permissionless and public global systems. Because this technology is often open-source, it is more likely than not that these types of digital object will proliferate over time, in number, use-case, design and technological functionality. If and when that happens, the law will find it increasingly difficult to maintain and uniformly apply rigid definitions to different types of technology or to different digital assets.

18. In anticipation of such difficulties, the Law Commission recommended that the Government create this Panel to provide non-binding guidance on the complex and evolving ways in which factual control can be exercised over third category things such as digital objects (and other issues relating to digital asset systems and markets more broadly). Crucially, in contrast to previous UKJT Legal Statements, such as those on Cryptoassets, Smart Contracts, and Digital Securities, this resource is less of a statement of the law and more of a practical explanatory guide, aimed at assisting Court users and the judiciary when considering how legal principles (specifically those set out in the Law Commission Report) can be applied *as a matter of fact* to various and evolving technologies. It became apparent to us during this Consultation that this was a point in need of clarification: this publication is not a statement or elaboration of legal principle or authority, but instead a practical guide and resource for those applying the law developed by the Courts following the Law Commission Report.

CHAPTER II: INTRODUCTION TO TECHNICAL ASPECTS OF CONTROL

19. In the sections below, the Panel seeks to describe the methods of control of digital assets, with the aim of helping a non-technical audience (comprised of the judiciary and Court users) refine their understanding of digital assets, to a sufficient degree so as to apply that understanding to legal analysis with which they may be faced.

20. The Panel has considered how best to present its technical discussion. There are broadly two ways in which this discussion could be presented. First, the Panel could describe the 'first principles' necessary to understand a digital asset system from scratch. However, the body of knowledge necessary – ranging from principles of computer science, cryptography, software engineering and so forth – is so vast that this approach is unlikely to be realistic or helpful to a non-technical audience.

21. Alternatively, the Panel could provide a brief explanation of some of the foundational principles, and then identify common real-world examples and the key concepts embodied by those examples, which are representative of situations that are either (i) most likely to reach the Courts and therefore fall to be considered by users and the judiciary; or (ii) are sufficiently general such that their explanation could help inform analysis of other situations, even if only by analogy. The Panel considers this to be more helpful and accessible, and thus more appropriate for present purposes (at least for its first output), even it is not as comprehensive as a 'first principles' explanation.

'Control' in a digital assets context

22. To situate the topic in its context, the question of control is perhaps best approached in stages, as follows.

23. First, imagine you are holding a valuable piece of jewellery in your hand. The question of who presently controls it is easy to answer: you do. It does not follow that you own the object, but it is clear that – absent the use of physical force – the party who can cause that object to end up in another's hands is you. If a Judge or other decision-maker with authority determines that the jewellery should be handed over to somebody else, they know who should be compelled and to whom any order to transfer should be addressed.

24. The question becomes more complex should the object in your hand be a ten-pound note. One ten-pound note is (generally speaking) fungible with any other, and this presents more options for a decision-maker ordering its transfer, because different alternative orders could be made for the transfer of funds which are not specific to the individual note in question. However, it is still fairly straightforward to identify the person who presently controls the sum of money which the note represents.

25. The analysis changes fundamentally when the question is concerned with ten pounds held in a bank account. The money may be held jointly by different account holders, whereby either party is authorised to instruct transfers out of the account. Alternatively, the account might be a long-term savings vehicle, whereby the owner of the money has agreed to 'lock in' the funds for a period of time in exchange for a higher interest rate, thereby relinquishing some control for that period of time, subject to the terms of the arrangement. Or, although the account is controlled on one level by the account holder (say, a firm of solicitors), the monies may be held on trust and to the order of the true owner (say, because they constitute client monies and belong to the solicitors' clients). There are many different permutations of forms of control.

26. Further complications are introduced if the bank account in question is accessible online. Electronic bank accounts introduce a new layer of control, whereby anyone who happens to know the password to an account and can navigate the relevant security protocols could be said to have control, regardless of who legally owns the account or whom an account holder has agreed with its bank should be entitled to operate it. Electronic bank accounts also often feature 'recovery' procedures, which allow account holders (e.g., those who have been 'locked out' of their accounts or fear their passwords have been leaked) to regain control and exclude others – meaning that the identity of those with control can change day-to-day. The question of who 'controls' money in an electronic bank account is thus not limited to those identified and agreed by a bank and its customers; it must take into account the reality of how a system operates, who possesses the information necessary to obtain control (by logging in or otherwise giving instruction), and the extent to which a bank could take exceptional action to exclude individuals, such as resetting a user's password.

27. Although electronic banking systems can be complex, their general operation is familiar to the judiciary and the wider public. Ultimately, an electronic bank account operates

using software and runs on computers controlled by the bank in question. That software is capable of analysis and its operator (the bank) is a known and identifiable entity. There are many tools that can help to resolve most real-world disputes that arise over control of bank accounts – e.g., transaction logs can be consulted, the owner of a particular bank account can be identified and contacted, a bank can be required to reverse a transaction, and so forth.

28. It is tempting to assume that the emergence of a digital assets industry does not fundamentally change such analysis, on the basis that a blockchain system is simply another form of software that can track value electronically and runs on physical computer hardware, much like a banking system. This is true to some extent, but there are critical differences that are important to understand and may become relevant in disputes involving digital assets.

29. Most importantly, one typically assumes that every banking system has an identifiable operator, which is usually the bank itself. This means that the simplest solution to most disputes over bank accounts is to require that operator to take a particular action, such as ordering a bank to move money from one customer's account to another, or to identify who gave an instruction to take a particular action on an account.

30. By contrast, in a blockchain-based digital asset network, the underlying system – the blockchain – is usually operated by a decentralised collection of globally-distributed parties, some or all of whom may be unknown or deliberately obscured. There may therefore be an absence of an 'operator' who could be compelled to take exceptional action to restore control for a rightful owner of an asset, or exclude others. As a result, in order to understand and resolve disputes surrounding control of digital assets, it is necessary for the decision-maker to understand the essential elements of how the system works and a digital asset is controlled.

CHAPTER III. FOUNDATIONAL CONCEPTS

What is a blockchain system?

31. To understand how digital assets are controlled in a blockchain-based system, it is necessary to understand the basic functions of that system. In this chapter, the Panel seeks to present the foundational concepts necessary to navigate discussion and analysis of such a system.

32. Imagine you are presented with a record purporting to represent a simple crypto currency transaction, in which Alice transfers 1 token to Bob. The following questions might arise:

    a. How can Alice prove that she was the party who authorised the transfer of the token?

    b. How can Bob prove that he now controls the token?

    c. Suppose Bob were to deny that he controlled the token received. How could anyone else prove Bob's control?

33. This may appear to be a simple scenario, but the correct answers are subtle and may be complex depending on the technology involved.

34. In this chapter, the Panel begins by analysing the model of a traditional banking system (as an example with which the readership should be familiar), then seeks to explore the conceptual differences and similarities in a typical blockchain system. It is hoped that the differences highlighted will help the reader understand a simple model for a blockchain system, and hence be better able to analyse situations such as that introduced in paragraph 32 above.

The crucial difference between blockchains and banking systems

35. A simple model for the technology (or 'technology stack') that supports a retail bank might include the following:

    a. First, there is a computer system that maintains a ledger;

b. Next, the ledger is governed by a 'business logic' – that is, a programmed set of rules and algorithms which dictate how the system should operate, e.g., by ensuring that a credit on the ledger is always matched by an offsetting debit;

c. The system is protected by access controls, which ensure the system's integrity by allowing only authorised users to perform certain actions; and

d. Finally, there is a set of procedures that dictates how the system evolves over time, as the number of transactions grows or in response to new business requirements.

36. Crucially, as noted in Chapter 2, it is often the case that every banking system (likely utilising proprietary or licensed software) has an identifiable operator – typically the bank itself. Given there is an identifiable operator, it is often unnecessary in the event of a dispute for litigants to understand in any real detail how the system works. The most straightforward solution to most disputes will be to require the bank to take a particular action. The bank may then be left to consider for itself what computer functions are necessary in order to implement the required action. The result is that a dispute between a customer and a bank can often be analysed purely through the lens of their legal relationship; how each manages their own books and records are matters which the Court rarely needs to understand or consider.

37. This is not necessarily so for disputes involving digital assets, because there may not be an identifiable 'operator' of a system who could be compelled to take any particular action. The lack of an operator with unilateral power to change a system's functionality is the defining characteristic of many blockchain systems. At its heart, it is that lack of a centralised operator which renders the control of digital assets novel.

"*Code is law*"

38. The lack of a centralised operator means that the only way in which a record could be made on the blockchain is to update it in accordance with the programming rules which govern the system. In other words, the only way to spend a particular quantity of (say) bitcoin would be to 'digitally sign' an electronic message (that is, a bitcoin transaction) with a particular 'private key'. If one does not have access to that key, then irrespective of who a litigant is or what a Court orders it to do, one cannot transfer a bitcoin and

there is no identifiable operator of the bitcoin system with privileged powers who could be compelled to transfer the bitcoin on another's behalf.

39. This concept is sometimes described as "*code is law*",[8] to mean that in blockchain systems, code determines whether a transaction will be recognised as valid by the network at the point of execution. That is not intended to mean that the legal status of an asset is defined by the state of the blockchain system; it does not follow that legal rights are defined by the blockchain state, nor that the system is immutable in all respects.[9]

40. Some aspects of the analysis may be different where a user is not actually exposed to the blockchain underlying a particular asset. Rather, his or her relationship may be with a service provider, such as a cryptocurrency exchange, and it is that service provider that interacts with the underlying blockchain on the user's behalf. In such cases, it is arguable that the existence of a blockchain system becomes less relevant, given a user has a legal relationship with an identifiable operator, and existing legal concepts regarding the relationship between an owner and a custodian might be more directly applicable.

41. In the Panel's view, it is those systems where a user directly interacts with the blockchain system without an intermediary service provider that introduce true potential novelty, and where an understanding of how the blockchain system operates becomes most important.

---

[8] The expression 'code is law', often associated with the work of Lawrence Lessig, is sometimes invoked in this context. Properly understood, Lessig's thesis was that software architecture can function as a form of regulation by structuring and constraining behaviour, alongside legal rules, market forces, and social norms.

[9] The reader should be aware, however, that there are some in the blockchain community who use 'code is law' as a normative concept, meaning that the code should be exhaustive in terms of controlling the asset and that traditional legal concepts should be displaced. (That is, there is no room for a Court to say that a transfer was illegal or lacked authority or was otherwise defeasible: the code itself is the final arbiter.) The 'DAO hack' incident on the Ethereum blockchain in 2016 provides a real-life example of this perspective. Somebody was able to drain significant funds from a smart contract through their close reading of the smart contract's source code and knowledge of how the underlying platform worked. In response, the majority of the community supported the execution of a 'hard fork' of the blockchain. The decision to override the "code" outcome through social consensus demonstrated that, even in permissionless systems, governance can supersede code. A parallel version of the Ethereum blockchain, which did not reverse this transaction, continues to run today, as Ethereum 'Classic'.

A simple model of a blockchain system

42. Imagine the existence of a novel, natively digital asset – something akin to bitcoin or ether. By 'natively digital', the Panel means something which exists only on a digital ledger, and is not linked to any real-world asset, including a claim or right of any sort to anything else.[10]

43. Next, assume that this ledger (which records the existence of this asset and its distribution amongst various accounts) is implemented as a publicly accessible shared spreadsheet.

44. A straightforward format for the spreadsheet would consist of just two columns, where each row records the name of an account (in the first column) and the amount of tokens held by it (second column). So, a 'payment' would consist of reducing a value in one row – the payer – and increasing the value in another row – the payee. The ledger below records Alice transferring 1 token to Bob, and Bob transferring 1 token to Charlie:

BEFORE:

Account	Ledger
Alice	3 token
Bob	2 token
Charlie	6 token

AFTER:

Account	Ledger	Notes
Alice	2 tokens	1 token transferred to Bob
Bob	2 tokens	Received 1 token from Alice; then transferred 1 token to Charlie
Charlie	7 tokens	Received 1 token from Bob

[10] This mental model is deliberately simplified, in at least two important ways. First, in real-world situations, it will often be helpful to distinguish between (i) protocol-native assets and (ii) application-layer crypto-tokens. A protocol-native asset (such as bitcoin/BTC on the Bitcoin network or ether/ETH on Ethereum) is intrinsic to the blockchain's consensus and economic security model. Its issuance, validation, and transfer are governed directly by the base-layer protocol rules, and it typically functions as block reward, transaction fee medium, or staking collateral. By contrast, crypto-tokens such as ERC-20 tokens or NFTs are created and governed by smart contracts deployed on top of a base protocol. They are not intrinsic to consensus and depend on the native asset and underlying protocol for their operation and security. Secondly, the 'ledger of account balances' model implied by the 'shared spreadsheet' metaphor maps well to some blockchain designs (e.g., Ethereum) but less well to others (e.g., Bitcoin), and the nuances of the underlying network can sometimes be important.

45. This spreadsheet (as with most mainstream blockchains) is fully public and can be accessed by anyone. To ensure its integrity, there must be rules governing how people can interact with the spreadsheet. Here, assume that anyone can attempt to update the spreadsheet, but their update will only be accepted if it meets two conditions:

    a. Rule 1: If one attempts to increase the value in one account, then they must reduce another account by the same value.[11] In other words, assets cannot be created or destroyed; they can only be moved between accounts.

    b. Rule 2: One can only decrease an account's value if one can prove that one is the rightful owner of that account or has been duly authorised by the owner. In other words, the 'payer' (the account holder whose balance is reduced) must authorise the transfer.

46. Assume further that the online spreadsheet is operated by a party, who is beyond the reach of any Court and is immune to reason: the rules above are inviolable and cannot be superseded.[12]

47. As to Rule 2, how does one prove ownership of an account in this context? A reasonable starting point might be to assume that given each account is secured by a password, whoever knows the password must therefore be the owner or have been authorised thereby. That is not sufficient in this context,[13] particularly given the blockchain is a fully public system and passwords are easily compromised with every use by which a user transmits the password (thereby revealing what it is to anyone with access). Instead, most blockchains employ a mathematical concept known as 'public key cryptography'. This technology makes possible a special kind of password – which we call a 'private key' – that can be used without being revealed.

---

[11] In the event that two conflicting transactions are submitted – e.g., Alice trying to spend the same tokens to two different recipients – we assume for simplicity that the operator approves at most one such transaction, and presents the same information to all users. Whilst this may seem to be a fundamental requirement of a transaction processing system, it is important to note that the decentralised nature of many public blockchain systems means there can be periods where it is uncertain which of two or more equally valid, but mutually conflicting, transactions will ultimately prevail. This is known as the 'double spending problem'.

[12] This is clearly an invalid assumption for a normal cloud service, but it is a key property of blockchain systems.

[13] Or, arguably, in any system: we only know that 'Open Sesame' opens the cave because Ali Baba overheard one of the forty thieves saying it!

48. The act of using a private key for the purpose of authorising a blockchain transaction is known as applying a 'digital signature'. This signature is a string of alpha-numerical digits, which is practically impossible to guess and can only have originated from the private key. It is valid only in the context of a specific transaction. This means the signature is safe to be transmitted publicly.

49. It suffices for the purpose of this Paper to know that a private key is a long, practically unguessable number (usually represented in alpha-numerical form). However, the box "*What is a 'private key'?*", below, offers more background on what a private key is and how they work.

50. Given that knowledge of a private key confers the power to sign transactions (i.e., spend an asset), it is very important that it is protected, both from exposure to others and from complete loss, which may render the asset in question inaccessible. For this reason, various techniques have evolved to protect private keys. In particular, physical devices such as 'hardware wallets' (often resembling a USB key) are increasingly popular. They often work by internally generating a random number from which a series of private keys can be generated, and converting this long number into a mnemonic phrase (a series of words, sometimes referred to as a 'seed phrase'), which can then be written down and stored in a safe place.

What is a 'private key'?

Pausing here, readers may wish to better understand what the Panel means by 'private key', which is a concept often used but perhaps less understood. It suffices for present purposes to state that a private key is simply a very long number, typically presented in alpha-numerical form. If a more detailed explanation is required:

Imagine a world in which nobody knew how to perform division or factorisation – i.e., if you were presented with the number 6, you are incapable of figuring out that it is 2 x 3.

Here is a game you could play with a friend, Alice, in such a world.

1. First, pick two numbers at random and in secret. Perhaps you pick 3 and 5. Keep these numbers to yourself for now.
2. Next, multiple your secret numbers (3 and 5) together, getting 15.
3. You share one of your numbers with Alice – let's say 5 – together with the product, 15.

4. You tell Alice, "Here is a number – 15. One of the factors is 5. I know the other factor, and I am going to prove this to you later". To repeat: you must assume that you and you alone know that the other number is 3 and that no one knows how to factorise numbers, such that Alice cannot figure out the other factor by herself.

How can you prove to Alice that you can factorise 15 without ever revealing that the other factor is 3? You could ask Alice to think of a random number (which, in the real world, might be a message that is being 'signed', e.g., a transaction). Let's say Alice's number is 13.

Alice could do two things. First, she could multiply the number (13) by the number she cannot factorise (15), to give 195. She keeps this number to herself.

Next, she knows that multiplying by a number is the same as multiplying by all its factors (e.g., multiplying a number by 6 is the same as multiplying by 2 and then multiplying the result by 3). She therefore takes her secret number (13), multiplies this by the factor of 15 which she does know (5), which yields 65. Anybody who knows the other factor could then multiply the result (65) by the secret factor (3), and the overall result will be the same as having multiplied 13 by 15.

Alice therefore shares this number – 65 – with you, and asks you to multiply it by the secret factor. If you know the remaining factor (3), then you should be able to calculate the same number Alice did when she multiplied the original number (13) by 15.

The only person who can perform the last step of the calculation above is somebody who knows the secret factor. To prove that it is you, you could multiply Alice's number (65) by the secret factor (3), which yields Alice's number (195).

Alice checks this against her own calculation (13 x 15 = 195), and can see that you have both calculated the same result. However, the only number she shared with you was 65; you were never told that her secret number was 13. So, you could not have calculated 195 in the same way she did: you could only have done it if you had knowledge of the secret factor of 15 (3).

Alice now knows that you must know the other secret factor to 15, and you have proved that you know this without revealing the secret number (3) to her.

The idea illustrated by this example, which underpins the use of private keys, is the principle that one can prove she knows things without revealing them. This is a foundational principle which underpins many public key cryptology systems.

51. The use of cryptology in this context is an extremely powerful mathematical capability, and it is used pervasively in distributed ledger technology ('DLT') and blockchain

systems.[14] However, just because a party knows or has access to a private key, that does not mean they are the legal owner of the asset associated with the key. A user may have gained access in an unlawful way (e.g., a hacker), or that user might otherwise not be recognised in law as the true owner (e.g., because the user acts as an agent or intermediary for the asset's true owner under a custody or trust arrangement). The blockchain system responds only to the correct use of a private key; it will blindly process a validly signed transaction irrespective of the identity of the signatory. Complications can also arise where more than one person could have access to a private key, wherein if a transaction has been validly signed for, it does not follow that that transaction has been authorised by a particular person, and it is in general impossible to know which holder of the key signed for any given message. In this way, it can sometimes be useful to think of assets secured solely by private keys as being akin to bearer instruments.

52. Returning to the questions at paragraph 32 above: if Alice wishes to transfer 1 token to Bob, she must persuade the blockchain system to reduce her token holdings by 1 and increase Bob's by 1. From a technological perspective, the blockchain does not distinguish between rightful or wrongful control – the system equates ownership and authority with the ability to digitally sign a transaction using the private key required. If the only available evidence is analysis of the blockchain itself, then all that can be discerned is that someone with access to the private key authorised the transfer – that is to say, the ability to cause a valid signature. Without more, the blockchain alone will not indicate how the transaction has been signed for (e.g., how a key has been accessed), and will not indicate who has accessed the key and authorised or implemented the transaction.

53. So far as Bob is concerned, he is always at liberty to demonstrate that he has access to a relevant private key, by simply authorising a spend of some tokens he controls. If the transaction is performed under the supervision of a trusted party – e.g., who dictates to Bob how much should be transferred in order to prove his identity – then it can be concluded that no other person who may also have knowledge of the key could have

[14] Note that we do not distinguish between 'DLT' and 'Blockchain' in most parts of this Paper, and so use them interchangeably to avoid unnecessary repetition of phrases. Where we intend to signal a difference, we do so explicitly.

performed the action, since they would not have known how much to transfer. This therefore proves Bob must have access to the key.

54. However, the position with third parties acting without Bob's cooperation is different. Nobody else can directly prove that Bob has the private key; they can only verify that fact upon Bob producing a valid signature. Cryptographic proof of control requires some action to be taken by Bob – e.g., by signing a message. Thus, without Bob's cooperation, the best evidence available will be circumstantial only, for instance by looking at associations or patterns of use. So, unless Bob decides to reveal his control by signing something with the private key, no one else can conclusively prove his control.[15]

Real-world applications

55. The key conceptual differences introduced in this Chapter between blockchain systems and traditional banking systems can be summarised as follows. Unlike a banking system, a decentralised blockchain system lacks a centralised operator which can implement extraordinary action (e.g., implementing or reversing a transfer between accounts without the cooperation of the account holder). In this sense, the system can only operate in accordance with its coding. Although the use of a secret PIN or 'password' (or similar security mechanism) for controlling a bank account may to some extent be analogous to the use of a private key, the operation of a private key which directly interacts with a blockchain is much more complex.

56. In the Panel's view, although the above introduction is a significant simplification of blockchain systems, it assists in setting out the basic concepts.

57. Of course, many situations with which the Court may be faced are not so straightforward. Today's blockchain systems can (and often do) deviate from the simple model above. In particular, although the use of a simple ledger of account balances helps understand the basic concepts, many popular blockchains are not implemented like this. For example, the bitcoin system is more akin to a ledger of physical cash transactions, wherein the amount being spent is often of a 'denomination'

[15] This insight also applies temporally: while physical possession tends to persist over time, even if it is only exercised at specific points, it is more difficult to see digital assets as being subject to control when nothing is happening to them. Control is generally only apparent when it is being exercised.

larger than the item being purchased, resulting in a need for 'change' to be returned to the purchaser (think of a £20 bank note being used to pay for a £19 transaction, with £1 being returned). By contrast, the Ethereum system more closely matches the 'shared spreadsheet' model when it comes to balance tracking.[16]

58. Further, as noted above, most blockchains will allow an asset to be controlled by more than one private key at the same time. For example, an asset could be configured such that either Alice's key or Bob's key would alone not be sufficient to authorise a transaction that spends it. Going further, some platforms enable assets to be placed under the combined technical control of two or more keys where different keys have different weights and/or keys from two or more distinct categories need to be used to authorise a transaction and so forth. Assets secured in this way are sometimes described as being under the control of a 'multisig' wallet, meaning that actions may require signatures from multiple keys in order to be valid. This is the subject of Chapter 5.

59. Until this point, the Panel has maintained the simplifying assumption that one can take a blockchain system as an immutable monolith: i.e., that its rules are never modified and operate as they were originally programmed. We relax this assumption in Chapter 6, in which we introduce the concepts of 'smart contracts' and 'protocol governance'. In brief:

    a. 'Smart contracts' concern situations in which the asset is further controlled by a computer program. Smart contracts are predefined pieces of code that are deployed on-chain, and can be used to enforce the rules governing an asset's lifecycle. In fact, smart contracts can extend their scope to implementing business logic, including governance mechanisms, identity management rules, interactions with other systems, or define complex financial instruments. This means it is often

[16] In some cases it may be necessary to consider how different blockchain architectures detect potential double-spends and hence preserve value as the system operates. In Bitcoin, each transaction consumes previously created outputs ('UTXOs') which, like physical cash notes, can each be spent only once by its owner. The protocol enforces this by requiring that every referenced output be unspent and authorised by the holder; once an output is included in a valid transaction, it cannot be reused. Any attempt to overspend, duplicate, or double spend an output is rejected by the consensus rules. During spending, a UTXO may be split into new outputs ('change'), but the sum of the new outputs must not exceed the value of the ones consumed, thereby ensuring conservation of value. Ethereum, by contrast, uses an account based model. Double spending is prevented through mandatory nonces attached to each account, ensuring that a payment instruction can only be processed once and cannot be replayed. A transaction is valid only if the account balance is sufficient and the nonce matches the expected sequential value. Attempts to repeat a transaction, alter its order, or spend more than the available balance will fail under the consensus rules. Although the two systems differ in structure, both achieve the same objective: ensuring that digital assets cannot be spent twice and that the ledger's integrity is maintained.

necessary to understand how a third party-authored extension to the platform works, in order to understand and answer questions of control. Note that smart contract control is not a replacement for, but in complement to, private key control: smart contracts can be used to enforce authoritative change of an asset's state, and are themselves controlled by administrators, who also possess relevant secret keys.

b. 'Protocol governance' concerns situations in which there may be processes that can be used to change how a system or a subsystem operates. There are many examples of widely adopted categories of products or services. It would be disproportionate to analyse them from first principles, so some common categories are described alongside the concepts on which they depend.

60. In Chapters 7 and 8, we survey specific examples of products and services categories that are concerned with notions of control, and so are likely to feature in any disputes related to this question.

61. Finally, in Chapter 9, we further relax our simplifying assumptions. The discussion preceding this point assumes that everything on a blockchain is fully visible to all. This assumption made it possible to assert that one can discern if a particular balance has been spent or if an account has received a payment. However, the use of various privacy-orientated technologies means these assumptions do not always hold true. We summarise some of the more common scenarios.

Further foundational concepts

62. To complete the introduction of foundational concepts, a general overview of current digital assets systems and the key terminology in the area is set out below.

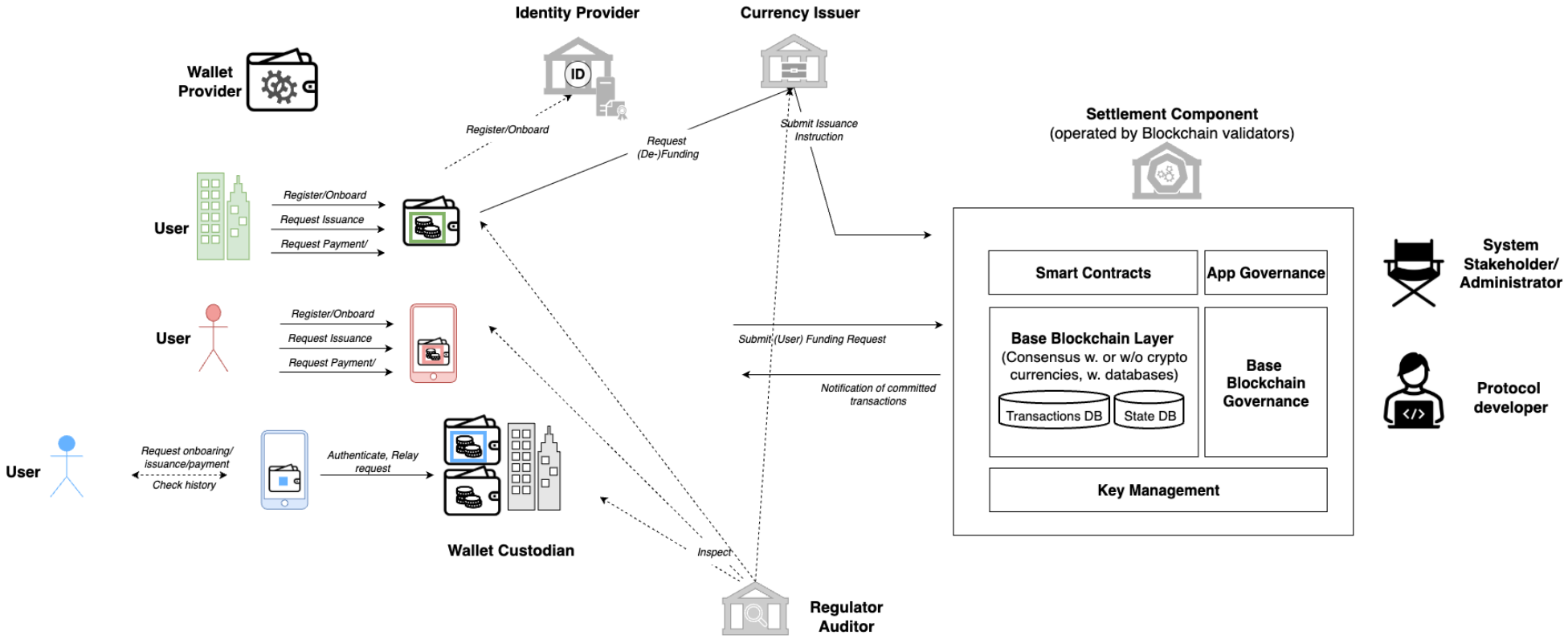

Figure 1: Overview of a digital asset system, where the asset is a currency.

63. Where the Panel refers to a digital asset system in this Paper, we refer to an electronic system that manages the lifecycle of an asset on a programmable and transparent transaction-processing technology. Figure 1 above aims to present, in diagram form, an overview of a digital asset system. The system features a currency issuer, a settlement component, one or more wallet providers, and (optionally) an identity provider and regulators/auditors. The system also includes the administrators of the different components and other players in the ecosystem (e.g., protocol developers).

64. A typical life cycle of a digital asset:

a. begins with the issuance of the asset (i.e., the introduction of the asset or new units of currency into a digital system);

b. continues through on-chain ownership transfers or delegated transfers, during which ownership of the asset changes as transactions are validated and finalised by the network, or ownership management is delegated to other users of the system or the system itself; and

c. where supported by the protocol, concludes with the asset's redemption or retirement, such as through burning or other on-chain mechanisms that remove it from circulation.

65. Generally speaking, a digital asset system is comprised of the following components:

a. **Asset issuer.** A digital asset is typically brought into existence by one or more asset issuers, who either jointly or independently put the new asset into circulation and manage it in many instances through a smart contract (e.g., Circle for USDC, Tether for USDT). Note that the identity of an asset issuer may not necessarily be known, and an issuer may not identify itself. Further, there are many examples of digital assets which, once issued, are entirely managed according to the logic of the blockchain system on which they were issued, and which have no linkage to any real-world asset (e.g., bitcoin).

What is a 'protocol'?

A protocol is a standardised set of rules which defines how a blockchain network operates. In this Paper, the Panel's references to certain 'protocols' might be better understood as

simply 'the rules which we take as given', in the sense that (for instance) the bitcoin protocol or the ethereum protocol operates in a certain way and one takes this as given. There may be processes by which the protocol can be analysed, debated, and updated, but in the context of a dispute (assuming, of course, that the dispute is not about the protocol itself), the Court is likely to simply be informed of what the protocol is and enquire no further. An analogy with traditional banking concepts might be a payment systems (e.g., the Faster Payment systems), which simply operates in accordance with a particular set of rules (e.g., certain institutions participate in the system, there are limits to the size of any given payment permitted, there are documents which specify the format in which instructions must be given, and so forth).

The foregoing is a simplifying explanation and the position in practice is usually more nuanced. Another way of conceptualising a 'protocol' is by reference to messages exchanged between the nodes of a system (e.g., bitcoin nodes or ethereum nodes) and the internal operation within those nodes, such that they reach consensus over the content of the generated blocks or the smart contract execution results. A 'protocol' ensures consistency in the rules applied across the nodes of a system, thus ensuring that they are all aligned on the version of the system being run.

b. **Protocol Developers.** Protocol developers build and maintain the core blockchain and smart contract infrastructure, which governs the asset system in question. For instance, popular systems such as Ethereum, Solana, Avalanche, and Cardano are developed, supported, and – to varying degrees – influenced by Ethereum Foundation, Solana Labs, Ava Labs, and Cardano Foundation respectively. It is important to emphasise that development of a blockchain protocol typically takes place in the open, allowing prospective consumers of the system or validators to do their own checks. From that point onwards, it is the validators or miners who, by choosing which versions of the underlying software to run and how they configure it, collectively determine the rules and emergent behaviour of the system: the foundations and firms supporting or influencing the platform cannot compel the adoption of any change they propose.

c. **End-user.** End-users are those who access their assets by installing wallets or using an access device of their preference. They can own and control assets, thereby contributing to an asset's lifecycle.

d. **Access device**. This refers to any hardware device on which an end-user can choose to install or operate their wallet software, and through which the user

interacts with a blockchain system. Access devices may include mobile phones, laptops or desktop computers, or enterprise managed machines.

e. **Wallet**. A wallet is a software application (e.g., like an app on a mobile phone), sometimes supported by a dedicated hardware component, that enables a user to generate, store and use private keys for signing transactions or proving ownership of digital assets. Wallet software manages transaction requests in accordance with the user's instructions while securely storing the private key or ensuring it remains within a protected execution environment. Wallets may take different forms, including:

   i. **Software wallets.** Applications (such as MetaMask) installed on a mobile phone, computer, or browser extension. These manage key generation and signing operations in software.

   ii. **Hardware wallets.** Dedicated physical devices (such as those manufactured by Ledger or Trezor) that store private keys in secure hardware and perform signing operations inside the device. Hardware wallet manufacturers do not hold, access, or manage users' private keys; they merely supply the device used for self-custodial storage.

   iii. **Custodial vs. non-custodial (self-hosted) wallets.** A custodial wallet is one in which a third party (such as a custodian or exchange) holds or manages the private keys on behalf of the user. A non-custodial or self-hosted wallet is one where the user retains exclusive control over their private keys, whether stored in software or hardware. In this report, the terms 'non-custodial' and 'self-hosted' are used interchangeably.

   Irrespective of the precise implementation, the defining function of a wallet is to provide a secure environment in which private keys are stored or accessed for transaction signing, without exposing those keys to unauthorised parties.

f. **Custodian**. A custodian is an entity engaged to hold or manage an end-user's digital assets or private keys on the user's behalf, thereby exercising control over the ability to authorise transactions. Custodians typically operate through specialised wallet infrastructure, enterprise key-management systems, or

institutional-grade operational controls. Custodians can be distinguished from other actors within the broader wallet ecosystem as follows:

i. 'Custodians' generally mean entities that hold or manage users' private keys and may sign transactions on their behalf.

ii. 'Custody/key management providers' (for example, Fireblocks) generally mean entities which supply secure key management technology used by custodians and institutions, but do not themselves act as a custodian for retail users.

iii. 'Self-hosted wallet software' (for example, Coinbase Wallet) generally mean arrangements whereby users retain control of their private keys and the provider does not hold or access them.

iv. 'Hardware wallet manufacturers' (for example, Ledger and Trezor) mean entities which design devices enabling self-custodial key storage. These manufacturers do not have access to a user's private keys and do not provide custodial services.

g. **Settlement regime.** This is a regime built into the relevant protocol, by which payments could be settled or transfers of asset could be requested in accordance with transactions that have been agreed on by end-users.

h. **Digital asset administrators / stakeholders.** These are entities responsible for defining and enforcing the rules of a digital asset system. Depending on the type of asset being considered, these stakeholders may include entities that authorise the issuance of assets, or entities that manage upgrades to smart contracts (that is, automated programs that execute asset-related business logic). In some cases, governance may be handled by decentralised autonomous organisations ("**DAOs**"), which are community-led groups that make decisions collectively through transparent, on-chain voting mechanisms. For privately issued digital currencies, the primary stakeholders are typically the institutions that issued the asset.

i. **Validators / Miners.** Validators and miners are participants in blockchain networks who help maintain the system's integrity by verifying transactions and reaching consensus on the current state of the ledger.

What is 'validation' and 'mining'?

In the Panel's view, a detailed understanding of validation and mining processes is not integral to understanding issues of control in a digital assets system. Nevertheless, the Panel offers below a short explanation, given these are commonly used terms which might arise in the course of a dispute.

'Validation' means the process by which a computer verifies the legitimacy of a transaction before it is added to the blockchain, e.g., by checking that a payer has sufficient funds to make the requested payment, or that the transaction has been correctly signed for using the required private key.

Broadly, blockchain protocols can be categorised as either 'permissioned' or 'permissionless'. In permissioned systems, participation is restricted to approved entities – examples include Hyperledger Besu, Hyperledger Fabric, and R3 Corda. By contrast, permissionless blockchains such as Ethereum and Bitcoin allow anyone to join, use, and participate without prior authorisation from a central authority.

In permissioned systems, validators are generally known and typically pre-approved. Transactions are validated often by consensus, using voting-based algorithms that allow collective decision-making among pre-approved participants.

By contrast, in permissionless systems, anyone can offer their services and participate as a validator or (as sometimes referred) a 'miner'. There are, commonly, two mechanisms of validation used:

1. **Proof of Work (PoW).** In PoW-based systems, miners use the computing power at their disposal to solve cryptographic puzzles. The first to solve the puzzle earns the right to add a new block of transactions to the blockchain, and receives a reward in the blockchain's native currency. By tying miners' work to rewards of cryptocurrency, the system incentivises miners to process transactions correctly. In this way, the system ensures that the required majority of computers (in the sense of accumulated computing power) participating in the network are behaving as intended, and the system as a whole remains secure and reliable.

   PoW is most famously used by Bitcoin, which was the first widely deployed system that enabled validators (miners) to participate in consensus without having to identify themselves or be formally admitted, thus unleashing the phenomenon of networks that operate without permission (anybody can contribute validator services) and, as a

result, operate beyond the control of any one identifiable central operator. It is this aspect - permissionless validation - that characterises the systems we focus on most in this Paper.

2. **Proof of Stake (PoS).** PoS-based systems operate in a similar way, save that validators are selected based on the amount of the blockchain's native currency which the validators own and have agreed to 'lock in' as collateral. As in PoW-based systems, these validators are responsible for adding and verifying new updates to the blockchain in exchange for a reward. Their financial stake serves as a direct incentive to act honestly; if a miner is found to have attempted to manipulate the system or failed to perform their duties, then they risk losing all or part of their staked assets.

j. **Regulators, auditors, and compliance entities**. These refer to those entities responsible for monitoring and enforcing legal standards, or performing audits on the various actors in the digital asset system. They play a particular important role in systems that feature centralised control or have fiat backing.

k. **Identity provider**. This optional component can be used by system stakeholders to provide authorised users with identities. These identities can then be leveraged to manage and assign different levels of access to system resources. The identity provider is also responsible for invalidating user identities once they are lost or compromised, or otherwise no longer valid, facilitate key rotation policies, etc.

66. A digital asset system which is built on blockchain technology is typically comprised of several technological layers, each responsible for different aspects of asset creation, control, and interaction. These technology stacks are explained further in this Paper, but at a glance:

a. **Base layer (Blockchain protocol).** At the foundation of the technology stack is the blockchain itself. As noted above, in general, one can distinguish between permissioned or permissionless blockchain protocol. In permissionless blockchains, the blockchain protocol is decentralised and maintained by a network of validators or miners. By contrast, in permissioned systems, the blockchain is usually maintained by a consortium or enterprise which retains a level of centralised administration and control. Whether maintained by way of open competition and punishment-and-reward systems (in the case of permissionless blockchains), or by way of governance frameworks (in the case of permissioned

blockchains), the integrity of a blockchain protocol's system of validation is central to the integrity of the system as a whole.

b. **Smart contract layer.** Built atop the base layer, smart contracts define how assets are issued, transferred, or controlled. These contracts can implement and enforce rules that have been agreed upon to govern an asset, such as vesting schedules,[17] multi-signature requirements, or automated economic operations (e.g., staking rewards,[18] buybacks,[19] etc). We distinguish between two prominent governance paradigms for smart contracts:

   i. Decentralised Governance, typically found in public blockchains, where smart contracts are governed by token holders or Decentralised Autonomous Organizations (DAOs). As noted above, DAOs are blockchain-native governance structures that operate without centralised leadership. They allow participants (usually token holders) to propose and vote on decisions such as contract upgrades, treasury allocations, or protocol changes. All actions are executed transparently via smart contracts, enabling collective decision-making without relying on a central authority.

   ii. Committee/consortium-based governance, typically found in permissioned blockchains, where the smart contract layer is managed by a set of authorised administrators, a relevant consortium or enterprise. Governance in these systems can be on-chain or off-chain, but follows predefined organizational policies or legal frameworks.

c. **Wallet and key management layer.** This layer enables end-users to interact with the system through cryptographic keys. As noted above, wallets can be a software (e.g., MetaMask, Phantom) or hardware (e.g., Ledger, Trezor) component. They can store private keys that are used to authorize transactions. Advanced

[17] This refers to schedules by which a token may be 'locked' for a set period, and then released gradually by reference to pre-set time periods.

[18] This refers to 'locking' cryptoassets in a Proof-of-Stake network to help validate transactions and secure the blockchain, earning rewards in return (i.e., new tokens or fees paid to stakers, miners, or holders for their support of a network.)

[19] This is when a project uses revenue to repurchase tokens from their market, reducing supply. This is similar to share buybacks in traditional finance.

wallet/custody solutions may use multi-party computation (MPC), threshold signatures, or social recovery mechanisms to enhance security and resilience.

d. **Data access and privacy layer.** Some systems, especially those focused on privacy (e.g., TEN or Zcash), include mechanisms to control visibility of transaction data. This may involve encryption, zero-knowledge proofs, or trusted execution environments. Developers can configure access policies to restrict who can view balances, transaction history, or event logs.

e. **Governance layer.** Governance mechanisms determine how upgrades, parameter changes, and administrative actions are authorised, and then implemented across the technology stack. Here, again there is a distinction between permissionless and permissioned systems:

   i. In permissionless systems, governance often involves numerous on-chain processes to ensure transparency, accountability, and community-driven decision-making. Some common processes include token-based voting (whereby participants use their tokens to vote on proposals, with voting power typically proportional to the number of tokens held); quorum requirements (which specify the minimum level of participation or agreement needed for a proposal to be considered valid); and timelocks (which introduce a delay between the approval of a proposal and its execution, allowing time for review or intervention if necessary).

   ii. In permissioned systems, governance is typically handled off-chain by a designated group – such as a board, foundation, or consortium administrators – who make decisions according to predefined policies or legal agreements.

A further word on governance

The Panel does not anticipate that detailed understanding of governance protocols is likely to be required to resolve the majority of digital assets disputes. However, readers may find it difficult to conceptualise how protocol governance operates in practice. An example may assist. In transitioning Ethereum to quantum resilient cryptography, the governance steps are as follows:

1. Ethereum researchers and the Ethereum Foundation explore the potential options for changes that can be implemented.

2. Community discussions take place on Ethereum research forums and governance channels, culminating in proposals being outlined.

3. Consensus is assessed off-chain. This might take the form of discussions with users or stakeholders, or using voting tools that weight votes.

4. If critical mass consensus is reached over a proposal, a prototype quantum-safe algorithm is introduced into the software. Tests are launched to validate the functionality and performance of the change.

5. Proposed implementations are then approved by the community (again, through e.g., off-chain discussions or using voting).

6. Ethereum nodes are then upgraded with the new algorithm.

CHAPTER IV. DIRECT OWNERSHIP AND CONTROL

67. The most straightforward method in which a digital asset might be held is through 'direct' ownership and control – i.e., control which is tied to a user through the possession of a private key, which is held by the user and accessible by him or her.[20] This method of control is, conceptually, not much different from the control of monies in a bank account through possession of the information necessary to 'log in' to the bank account in issue or navigate a bank's security protocol.

68. However, in the context of resolving disputes, a key issue concerns the method through which users typically store their private keys, which might render them susceptible to theft, tampering, or interference. This Chapter is therefore focused on how private keys are typically held (i.e., using wallets), and the technologies widely used to secure or recover them.

69. In Chapter 3 above, the Panel introduced the concept of a wallet. Many users manage their assets through a self-hosted or non-custodial wallets. Typically, such wallets generate and store private keys 'locally', on a hardware device or computer of the user's choice. Such 'local' storage, which is usually tied to a hardware device, helps ensure the security of a key so long as that hardware device can be protected. However, this also means that if the device is lost or tampered with, and there is no other means in place to restore access to the private key,[21] then access to the relevant digital asset is permanently lost.

70. To address these risks, hardware wallet providers have introduced technologies that help improve security of the hardware device. At a high level, it is possible to distinguish between such technologies, which fall into several categories:

---

[20] For the avoidance of doubt, the mere possession or control of a private key constitutes evidence of the ability to effect transfers (i.e., spending authority) over a digital asset, but does not necessarily establish legal ownership of that asset. Control of a private key may be exercised by a person who is not the rightful owner—for example: (i) where a private key has been compromised or stolen; (ii) where an employee uses a company-held key solely as an agent of the company; or (iii) where recovery phrases, backups, or shards are held by custodians, service providers, or other third parties pursuant to contractual or technical arrangements.

[21] Frequently, users installing a wallet application are prompted to choose a seedphrase, i.e., a 12-24 word phrase they can memorize (or store offline), that is used to generate that wallet's secret key. The seedphrase can be used to regenerate the wallet's key in cases where the wallet device is lost. Where we refer to permanent loss of the private key in this Report, we include concurrent loss of any associated seedphrase.

a. **Tamper-resistant hardware**. There are technologies which are designed to generate and store keys within a tamper-resistant hardware device, such as Trezor and Ledger wallets. Generally speaking, these devices are designed such that if a physical attacker attempts to access the device to extract a key, then the key is automatically deleted to prevent it from being compromised.

b. **Key-wrapping.** Another popular technique is to back-up a private key after encrypting it with another key, such that a key cannot be accessed without access to a second, a master key.

c. **Trusted execution environments ('TEEs') and Hardware Security Modules (HSMs).** TEEs are hardware technologies that can be configured to run specific pieces of code to better guarantee security. HSMs are hardware devices with the ability to store sensitive pieces of information (e.g., secret keys) and use it for restricted operations inside the device (e.g., cryptographic signing) in a tamper-resistant manner. TEEs are typically combined with HSMs to fortify user wallets. In broad terms, when a user initiates a transaction, the wallet signs for it internally and only the signed transaction is transmitted out of the device – the private key itself never leaves the device. This architecture reduces the risk that a private key may become exposed even if a connected computer or mobile device is compromised.

d. **Offline signing technology.** More advanced security measures may include storing the key in tamper-proof, offline devices, which only become online (and accessible) according to a predefined schedule; an example is IBM's Offline Signing Orchestrator, which operates as follows. Any changes to this schedule require a multi-party approval as per a system-wide policy, which adds an additional layer of protection. During offline periods, the device remains completely isolated and inaccessible via any network or other system components. Then, at specific, controlled times, the device is temporarily connected to internal components that route signing requests to it. These requests are processed entirely within the device (providing a TEE, explained above), and the signed output is returned to these internal components, which further relays the request to the rest of the system.

This type of mechanism significantly reduces the exposure of sensitive key material and their signing capabilities to attackers, because the key is exposed only during specified signing windows and thus any attack during offline periods will be delayed before them taking effect. This renders it more likely that an attack will be detected before an attacker gains access to a signature.

71. Although an understanding of how a non-custodial or self-hosted wallet operates is integral to understanding how digital assets are controlled, in practice many digital asset owners use custodial (or 'hosted') wallets, which are managed by a third party service provider (such as cryptocurrency exchanges or banks with a custody offering) which hold users' private keys and facilitate transactions on their behalf. The end-user entrusts the security provider with the execution of the transactions which he or she has authorised and ensuring they take effect on the blockchain.

72. The additional layer of control introduced by such custodial wallets is discussed in Chapter 7 below. For present purposes, it suffices to note that so far as security is concerned, a custodial wallet typically (i) makes use of TEEs; and (ii) relies on traditional authentication mechanisms for verifying user's requests (e.g., single or multi-factor authentication, not dissimilar to security mechanisms that might be used to enable remote log-in to a network or verify a bank transaction). It should be emphasised that how a third-party custodian or exchange operates its custodial wallets will need to be examined on a case-by-case basis. However, fundamentally and whether a wallet is self-hosted or hosted, the chief aim is to ensure the protection of a user's private key.

73. The discussion up to this point has focused on the existence and possession of a hardware device. Devices can be misplaced, lost, or stolen. They can also be susceptible to hacking, whereby a hacker could gain access to the software component that is authorised to request signatures from a hardware device (so, even if the hacker has no physical access to the device, it could manipulate the device into producing a digital signature that could enable a hacker to impersonate the owner of the device and make a request to the blockchain).

74. To address these issues, a number of solutions have been developed, such as social recovery wallets (e.g., Ready) and multi-party-computation ('MPC') wallets (e.g., ZenGo). The control of an asset using these technologies is still determined by the

possession of a secret key, but the key is distributed across multiple participants, such that cooperation between participants is necessary to produce a valid signature on a given transaction. Imagine, for instance, that a secret key ('SK') could be divided into four component parts (SK1, SK2, SK3, and SK4), and that each is distributed to different users (U1, U2, U3, and U4). In a '3-out-of-4' policy, three of U1 to U4 would need to agree on a given request for a valid signature to be produced, which reduces the risk to an asset if one user has been hacked and his or her access has been compromised. In such cases, a user's control of an asset through the use of a private key requires the cooperative action of other users.

75. Thus far, the Panel's discussion of direct control has been based on the assumption that the control of a digital asset is contingent upon possession of a private key which can authorise transactions associated with that asset. Although this will be true of many digital assets, to complete the analysis, we can relax this assumption by highlighting three exceptions:

a. **Identity-based systems**. In permissioned systems, rather than being dependent on the possession of a private key, control can be identity-based and can be managed through digital identity certificates or credentials. In these systems, there is typically a centralised operator or institution, which has authority to grant, renew, or revoke a user's identity credentials. Here, a user's control is determined by the possession of a particular user's key, as well as the intervention of a centralised network operator with authority to verify users' credentials. Or, put another way, 'Rule 2' at paragraph 45 above is determined by both the possession of a private key, and also an administrator being satisfied that the user has the necessary permissions in place in order to deal with the asset. Examples of such systems include Hyperledger Besu, Hyperledger Fabric, or R3 Corda.

b. **Stablecoins**. Stablecoins systems, such as USDC or USDT, are an exception to the general model of digital assets discussed up to this point. As they are asset-backed tokens (meaning, they represent claims in off-chain, fiat currency), and although users can self-host stablecoin, stablecoin issuers (Circle and Tether, respectively) retain administrative privileges that allow them to freeze or revoke access to assets, typically in response to legal or regulatory requests. Given the existence of a centralised operator, recovery of assets is typically not user-driven

but may be facilitated by the issuer under specific conditions, such as fraud or loss of access credentials. A stablecoin holder's factual control is always subject to the overriding authority of the issuer.

c. **AI agents**. The emergence of AI agents that autonomously transact with digital assets has led to the creation of specific frameworks in which they can operate. For example, the Ethereum community has developed ERC-8004, a standard for how AI agents are identified, vouched for and validated on-chain. However, ERC-8004 does not give AI agents cryptographic authority over digital assets nor define how AI agents control assets. Actual authority over digital assets still depends on wallet, account and smart contract designs. Thus far in this Paper we have assumed control is exercised ultimately by a human, whether via an agreement, a legal relationship, or independently. However, increasingly AI agents are capable of being granted access to private keys and executing transactions with varying degrees of automation and removal of checks by a human.

CHAPTER V. MULTI-PARTY AND SHARED CONTROL

76. In this Chapter, we expand on the use case explained in Chapter 4 by introducing the concept of multi-party or shared control. These refer to related but different concepts:

a. '**Multi-party control**' refers to situations in which an asset is controlled by multiple different parties, who may all be independent from one another, but not necessarily. In this case, no individual party can unilaterally exercise control. A decision about the asset will require agreement between all or some of them. The rules for how agreement is reached are defined by a policy – e.g., the governing policy might require unanimous agreement (all parties must approve) or quorum agreement (e.g., approval by a majority or pre-determined number of parties).

b. By contrast, '**shared control**' typically refers to situations where any one party among a group can act independently so as to control the asset, without needing approval from others in the group.

77. When considering disputes concerning such use cases, Court users and the judiciary may encounter discussions about 'trust assumptions' – that is, how much the parties within a group trust one another and what levels of compromise will be tolerated in the system. For instance, if the system assumes that no more than b parties can be compromised or fail at the same time, then it should require that $b + 1$ parties approve any action. This ensures that even if b parties' access is compromised or they are otherwise unavailable, the system is still capable of being operated. (Relatedly, the Court may also encounter discussion of '*t-out-of-n*' policies. These typically refer to situations where t represents the minimum number of approvals required to operate an asset out of a total of n parties, whereby t will be set to $b + 1$.)

78. As in Chapter 4, the Panel anticipates that most disputes concerning assets under multi-party control will focus on the mechanism by which such control is being enacted. The most widely adopted approaches to enable multi-party control over digital assets:

a. **Multi-signature wallets.** These technologies allow multiple independent users' wallets to co-sign a transaction request, thus avoiding reliance on a single party and reducing the risk of a security breach compromising a user's wallet. A key advantage of these wallets is that different parties can each use their existing

cryptographic keys without needing to generate special-purpose keys specifically for the purpose of multi-signing. However, to create a valid multi-sig transaction, the wallets must coordinate off-chain, meaning they interact outside of the blockchain (for instance, through a secure messaging protocol) to assemble the required signatures. The system (typically a smart contract) that receives the final request must be programmed to verify that the correct set of signatures is present.

b. **Multi-party computation ('MPC') based wallets.** Under this approach, the private key that controls the asset in question is never held by a single party. Instead, it is split into multiple shares held by different parties, with those shares being generated using cryptographic techniques that respect the desired threshold policy (i.e., so that *t* out of *n* parties can jointly produce a valid signature). This approach is most common in custodial or institutional settings. However, depending on the specific MPC scheme used, note that the key shares may not conform to standard cryptographic formats (referred to as 'non-standardised key materials') and may not be compatible with certain hardware, which can complicate secure storage and interoperability. The U.S. National Institute of Standards and Technology is currently considering potential schemes for standardisation, so this remains a developing area.

79. The parties to a multi-signature or an MPC-based signature can leverage a coordinator or a smart contract to orchestrate their actions and collect the signatures necessary for transaction execution. This can take different forms:

a. **Coordinator-based orchestration and signature collection.** Under this approach, the parties usually designate a coordinator whose aim is to collect and combine the requisite signatures. While the coordinator can drop signatures and not include them in the transaction, it cannot exercise control by itself; it still needs to collect signatures from enough parties to meet the policy governing multi-party control.

b. **Smart contract-based signature collection.** Under this approach, the smart contract itself enforces the relevant threshold policy governing multi-party control. Each party independently submits a signed transaction to the blockchain (indicating their approval of a particular transaction); the contract monitors such submissions, and executes the transaction only once the required number of approvals is reached.

This model is most common among DAOs, institutional custody platforms, and high-value asset management, where operational resilience, transparency and auditability tend to be most critical.

80. The commonality between such systems is that they are designed to prioritise resilience and fault tolerance; that is, if one party loses access to their key or device, the system can still function so long as the threshold of approvals is met. Some systems also support contingency mechanisms such as key rotation via trusted third-party guardians, which is intended to further improve resilience.[22]

81. The above mechanisms can be present in both permissionless and permissioned systems. However, an additional layer of nuance might be present in permissioned systems, whereby users tend to be uniquely identified through a verifiable identity issued by a centralised authority, such as an administrator or a certificate authority, which will typically enforce access control policies that pre-define who is allowed to perform what actions on a given asset. Control will typically be implemented through workflow-based approval (e.g., a transaction may require sign-off from an initiator, a reviewer, and an approver before it is executed). This could be expressed cryptographically using multi-signatures (e.g., a signature from the initiator, another from the reviewer and a third from the approver). There are cases where workflow approvals are not expressed cryptographically or even digitally, and control is therefore not confined to parties holding cryptographic keys authorising transactions. As discussed further in Chapter 8 below, where privacy may be a concern, there are protocols which allow multi-party control without revealing whom among a group of parties approved an action (e.g., where vote confidentiality or anonymity is important).

82. It should also be borne in mind that given permissioned systems are centrally coordinated, there may be mechanisms available for overriding the relevant controls. That is typical of stablecoin issuers, who commonly employ multi-party controls and require multi-signature approvals (particularly for the freezing of assets or executing

[22] Two instances of key rotation can be distinguished. The first is when key rotation is the result of key expiration, which can be addressed before key expiration by transferring the control of the asset from the soon-to-be-expired key to the new key. The second is when key rotation is triggered after a security breach, which requires override rights, exercised by stablecoin issuers to transfer control from the compromised key to the new key. In this case, the stablecoin issuers may need to verify that the override request originates from the rightful party.

large transfers). It is also common for such issuers to have a centralised governance multi-sig (e.g., a ‘security council’) who can intervene in emergencies or enforce regulatory actions (e.g., putting in place trade restrictions to comply with sanctions regulation).

CHAPTER VI. SMART CONTRACT-BASED CONTROL

83. As noted above, a smart contract is a predefined piece of code that enforces rules governing an asset's lifecycle. Those rules may include the conditions that need to be met for an asset to be transferred, or some other action to be orchestrated (such as governance actions, or compliance checks). Smart contracts frequently implement not only transactional logic, but also administrative and governance controls that shape how future rules may be modified.

84. In the preceding chapters, the Panel discussed forms of direct or multi-party ownership and control of digital assets. Frequently, smart contracts are not a conceptually distinct form of control, but rather a mechanism by which those forms of control are enforced in practice. That is, a smart contract can be programmed to only allow a requested action to take place (e.g., a change of ownership) and update the system's state accordingly if the requestor(s) comply with a certain policy.

A basic smart contract system

85. The commonality between all smart contracts, whether in permissionless or permissioned systems, is that they execute predefined logic. At its most basic, smart contracts can be used to hold and manage assets, encoding the conditions under which an asset is permitted to change ownership. The encoded options vary significantly, and can range from requiring just a signature from the asset's owner, to requiring multi-party signatures or the satisfaction of numerous other conditions (e.g., loan repayment, auction completion, or more simplistically, an asset is only permitted to move from A to B if B has paid $5 to A). This constitutes smart contract code execution control, meaning the contract enforces whatever logic has been encoded.

86. Further, smart contracts can also be programmed to decide and trigger asset ownership change. The specific smart contract to be executed will depend on the type of asset in question and the context of the transaction, but by way of example, decentralised exchange ('DEX') smart contracts allow users to trade assets (or tokens) directly without intermediaries, automatically updating asset ownership when trades are executed. They can implement sophisticated logic, such as automated market-making algorithms, liquidity pool management, and price discovery based on supply-demand dynamics which the smart contract monitors. The ability of such contracts to handle

transactions autonomously helps to ensure trustless and transparent asset transfers between participants.

87. In some use cases, asset owners (in this context, known as liquidity providers) can delegate ownership of their assets to a smart contract (in this context, known as the liquidity pool), usually in exchange for a fee. Liquidity providers thereby 'lend' liquidity to the smart contract, through the transactions they submit to the blockchain system referencing their assets. From that point onwards, the assets can be traded on the blockchain autonomously. DEX clients typically request exchange of an asset they own, with an asset in the liquidity pool, assuming an exchange rate range. Such requests can then be handled directly by the smart contracts, which can facilitate the change of ownership of the relevant assets and fee distribution across liquidity providers. Underpinning these flows is an implicit trust that the liquidity providers place in the liquidity pools and the associated market-making algorithms underlying the smart contracts (and its correct implementation).[23]

88. Trust in smart contracts derives from the adoption of well-established standards for common asset lifecycle management operations and from the way in which smart contracts and the underlying blockchain protocols are governed. By the word 'standards' in this context, the Panel means community-defined specifications that ensure interoperability and predictable behaviour. For example, Ethereum's "ERC" (Ethereum Request for Comments) standards define how tokens should behave on-chain.[24]

Layered forms of smart-contract-based control

89. Although some smart contracts operate fully autonomously, in practice many systems deploy a layered form of control which is enabled through smart contracts and underlying blockchain system. In the Panel's view, given the proliferation of such systems, Court users and the judiciary are highly likely to encounter smart contracts

[23] However, control in such cases may be still layered: a DEX contract may contain administrative keys, upgradability mechanisms, or oracle dependencies affecting price data. Thus, while execution appears automatic, identifiable actors may retain override, pause, or modification powers.

[24] Such standards typically define only execution behaviour, not the broader question of who can modify or influence the contract's functioning.

that feature multiple overlapping control layers rather than simple, code-only execution in the course of resolving digital assets disputes. It may therefore assist readers to be aware that the following layers of control typically feature in a smart contract's operation (though this is, and can only be, a starting point for examining specific smart contract systems):

Signing control

90. This refers to control arising from the possession of private keys required to authorise transactions, which has been discussed earlier. As explained above, participants in permissioned systems will typically be identity-authenticated (typically through digital certificates issued by a trusted authority), meaning that transactions are usually linked to a known individual user or organisation, rather than to an anonymous wallet address. In this context, smart contracts may play the role of enforcing role-based permissions (i.e., ensuring that only certain authorised users are allowed to perform specific actions), and may be deployed to enforce identity management to help to ensure traceability and auditability as necessary (or required as a matter of regulation).

Programmatic (Deterministic) Execution Control

91. Programmatic control refers to control embedded in the contract code, which governs when and how assets may move and system state transitions occur (e.g., adapting account balances). Where contracts are immutable and lack administrative override, practical control may be limited to this layer: that is, once deployed, the contract executes its logic as written. Programmatic control should therefore be distinguished from administrative control (discussed below), which can change the code that will be executed or interrupt/modify its execution conditions.

Administrative control

92. Administrative control refers to the ability to upgrade, pause, freeze, or otherwise modify the contract's functioning (e.g., via admin keys, multisig arrangements, timelocks, or emergency guardians).

93. In the discussion thus far, for ease of exposition, the Panel has presented a picture of permissionless public blockchains as immutable. We can relax this assumption by

noting that smart contracts can be designed in a way which enables modification, for instance to perform security upgrades. Because of the potential ability for smart contracts to modify an underlying blockchain, how upgrade policies are set and enforced are critical from a security and control perspective. Indeed, the changeability of a smart contract may potentially be relevant to legal analysis, as such changeability may also mean that future conditions of control might be susceptible to change.

94. A common approach to management administrative control is to implement access controls restricting administrative actions to privileged accounts (e.g., multisig wallets) and/or to smart-contract-based governance (e.g., DAOs that collect votes and execute changes upon meeting preset thresholds). Additional safeguards may also include timelocks (delaying the effect of changes to allow review[25]), emergency guardians (with narrowly scoped pause/freeze powers for incident response[26]), and formal review or rollback windows.[27]

95. In this context, permissionless and permissioned systems typically have different features. In permissioned settings, smart contracts are generally upgradeable in accordance with the administrator's policies. The ability to modify or override contract behaviour is therefore reserved to authorised administrators or governance committees, and typically subject to formal review, audit, and compliance processes.

96. Similarly, stablecoin contracts usually feature tightly administered controls. Note that for stablecoins, smart contracts tend to be tightly controlled by issuers, and often

---

[25] Meaning, mechanisms that delay the effectiveness of administrative actions by imposing a mandatory waiting period (e.g., 24–72 hours). During this interval, users and auditors can review the proposed change and, if necessary, withdraw funds or object through available governance channels. Timelocks therefore serve as an *ex ante* safeguard against the unilateral or instantaneous exercise of administrative powers.

[26] Also sometimes referred to as 'pause guardians', 'security councils', or 'emergency multisigs'. These are entities that are granted narrow powers (typically the ability to pause or freeze critical contract functions) in response to security incidents, oracle failures, or ongoing exploits. Their authority is intentionally limited (for example, they may pause functions but cannot upgrade logic or transfer assets) and often subject to sunset clauses or subsequent governance ratification.

[27] Meaning processes, contractual features, or governance rules allowing post-implementation scrutiny of an upgrade, and in some cases permit the reversal of a change (e.g., by reverting a proxy implementation to the prior version) if unexpected behaviour, vulnerabilities, or non-compliant outcomes are detected. These mechanisms provide an ex-post safeguard, complementing timelocks and emergency powers by ensuring administrative changes remain auditable and correctable.

include functions for minting,[28] burning,[29] or freezing or blacklisting users and their access.

97. By contrast, permissionless systems can typically be distinguished into two categories:

    a. **Immutable contracts** (e.g., early Uniswap v1 pools), which contain no upgrade path or admin key. Once deployed, their logic cannot be altered. Control is limited to programmatic execution under fixed rules.

    b. **Mutable/upgradeable contracts** (common in contemporary DeFi), which employ proxy architectures allowing logic replacement while preserving stored state (e.g., user balances). For example, in typical lending protocols, users deposit assets into a contract while a governance multisig retains overarching upgrade and parameter authority. That multisig may adjust liquidation thresholds, pause withdrawals, or modify risk models.

Governance Control

98. Governance control refers to mechanisms by which who has administrative authority and under what conditions can itself be changed (for instance, through changes in token-holder voting, DAO, or appointed committees procedures). Smart contracts can be programmed in order to increase or reduce the powers of an administrator, rotate signers of a multisig, alter timelock durations, or install/remove an emergency guardian.

Oracle/Data Control

99. Smart contracts sometimes rely on external data feeds (or 'oracles') that provide off-chain information such as asset prices or real-world event outcomes. Data provided by these oracles is used by smart contracts to determine whether something should be executed by the contract. Even though oracle providers do not have control over assets, their services are relied upon to execute actions on assets. This has led to oracle manipulation, which could potentially be used as an attack vector by nefarious actors to force a specific smart contract execution.

---

[28] This refers to the creation of new tokens.

[29] This refers to the permanent destruction of tokens.

Blockchain Protocol Control

100. As noted at Chapter 3 above, 'protocol governance' concerns situations where a system or a subsystem can be changed. Smart contract upgrades are not the only way in which systems can be changed. Protocol governance concerns changes at the base layer (e.g., consensus rules, virtual machine behavior, fee markets) that may affect contract execution, finality, and security, as well as upgrade pathways for deployed contracts.

101. Note, however, that there are likely to be significant differences between permissionless and permissioned systems in this regard:

a. In permissionless systems, control at the protocol level is usually exercised through a form of decentralised consensus mechanism – such as Proof of Work or Proof of Stake (see Chapter 3 above). These mechanisms are designed to be open, meaning that anyone can in principle participate. However, the ability to change how the protocol itself operates is typically managed through off-chain or on-chain voting systems, which involve participation and consensus among token holders, core developers, or DAOs. As a result, protocol governance in public blockchains tends to be slow. Notwithstanding that, changes (when they take place) have the potential to fundamentally redefine how smart contracts operate and how users manage their assets in any particular system; it is therefore important to ensure agreement between stakeholders before changes are implemented.

b. By contrast, in permissioned systems, protocol-level control will typically be handled centrally by an administrator. As noted above, because governance in permissioned systems is tightly coupled with identity and access management, only authenticated participants can propose and/or approve protocol changes, and such changes are usually subject to formal review and audit. This allows organisations to align protocol behaviour with operational policies, regulatory mandates, and risk management frameworks.

CHAPTER VII. DELEGATED CONTROL

102. For ease of exposition, much of the discussion up to this point describes a 'purist' form of control, whereby individuals interact directly with blockchains and take responsibility for their own safety without using the services of third parties. To the extent additional protections were required, the Panel has described means of implementing protection through 'on chain' techniques (such as multi-signature schemes in Chapter 4) or business rules encoded programmatically (Chapter 5).

103. However, the reality is that not everybody who deals with digital assets will have the skill or appetite to take personal responsibility for securing such assets. Numerous firms have emerged to perform this work on those users' behalf.

104. In principle, such use cases may be easiest for the Court to analyse as they are most analogous to existing systems in which there is a contract between a user and a service provider which can be interrogated. However, the terminology deployed can often obscure the underlying simplicity of a system. In this Chapter, the Panel aims to unpack the prevailing terminology and basic structures.

105. The Panel discusses three main scenarios of delegated control below. Whilst the three scenarios are not comprehensive, the Panel hopes that they will provide a starting framework for approaching a dispute involving delegated approach.

Crypto exchanges

106. Many people first interact with public blockchains through an online 'crypto exchange'. Firms such as Coinbase provide a user-friendly website and app which allows users to buy and sell digital assets, hold them securely, and sometimes even invest or lend them on the user's behalf. In the simplest scenarios, a user will fund purchases with fiat currency through a bank account or payment card, and manage assets through a website or an app; that is, the user has no direct interaction with a blockchain of any sort.

107. In this context, it is important to note that the term 'exchange' can be misleading. Unlike in traditional finance where a stock exchange has a narrowly defined meaning (which is clearly differentiated, for example, from concepts such as brokerage, custody, clearing, or investment management), crypto exchanges are often vertically integrated and routinely perform some or all such functions.

108. Thus, when analysing situations involving crypto exchanges or similar entities, the Court and litigants should seek to identify precisely what services the crypto exchange provides and how such services envisage implementing asset control. For instance, where a customer 'buys' a digital asset on a crypto exchange, often the exchange operator has simply recorded an entry as a credit to the account holder, being a liability to transfer the relevant amount of the digital asset to a wallet of that customer's choice on demand in the future, and which involves no use of any blockchain until that time. This may seem counterintuitive (one would typically think of 'buying' an asset from a seller as meaning that seller (the 'exchange' in this case) thereby holds that asset on your behalf), but it may arise from the fact that some digital assets exist on more than one blockchain – e.g., many stablecoins are available on multiple chains. Thus, when a user purchases a stablecoin on an exchange, that user may not yet know on which underlying blockchain that stablecoin will ultimately be used, and the user may not be interested in anything but a future entitlement on demand. It is only if that user instructs the exchange to transfer the asset to a wallet of his or her choice that it will be necessary to transfer ownership.

109. Similarly, in many cases the payment into an exchange account or wallet of cash (or digital assets) sees fungible assets pooled and added to the exchange's liquidity pool, and the account holder is given an 'IOU' on their account. The balances displayed on any account are representatives of the value of asset(s) owed under the IOU, but are often used by the exchange themselves (similar to fractional banking). Upon the customer's instruction to transfer funds out of the exchange, the exchange will use funds from its liquidity pool to execute the transaction on chain.

Crypto custody

110. The foregoing paragraphs describe a situation where a user's interaction with an exchange is conducted entirely through the exchange's website or app. We can relax this assumption by considering a user who already owns some digital assets held in a wallet, and who wishes to enlist the services of a third party to provide safekeeping services. Some crypto exchanges will provide this service, but there are also specialist 'crypto custodian' service providers.

111. To explain such use cases, consider first a simplified model and a more sophisticated variant:

a. **The simplified model.** A common model of crypto custody is one where a user logs into a provider's website and requests a wallet 'address' to which he or she can send the digital asset which he or she wishes to be held in custody. The user then sends his or her assets to that address, thereby transferring control to the custodian. Once received, such assets become subject to the control of a private key known only to the custodian and not the customer. It is the custodian, not the customer, who has the technical ability to authorise transactions using the underlying tokens. (Numerous important questions may arise in such scenarios, which will depend on the terms agreed upon by the custodian and the customer: e.g., whether the assets are ringfenced (as though, for instance, it were client money held by a solicitor), whether the assets are treated as fungible such that the custodian has only a liability to deliver an asset of equivalent value on demand without expectation that the exact same physical coins and notes would be returned, etc.)

b. **Crypto custody as key management.** In a second, more sophisticated variant of the above model, the owner of a digital asset utilises a technical feature of the underlying blockchain platform to add the custodian to the set of parties required to authorise certain actions on the asset – typically the 'multi-sig' concept described in Chapter 5. All of the analysis in Chapter 5 applies to such key management services and it is important to interrogate the terms on which such control will be implemented. For example, if the asset is secured by a '*2 out of 2 multisig*', then the likely intention is that *both* the original owner and the custodian must approve an operation before it can be performed. (Here, a key question will be under what circumstances should or will a custodian countersign a transaction presented to it, and what due diligence (if any) is it expected to perform?). By contrast, if the asset is secured by a '*2 out of 3 multisig'*, then the likely intention is to enable two people (say, a couple) to co-own an asset and to involve the possessor of a third key (the custodian, in this case) to decide which party's will should prevail in the event of a dispute. Such use cases may be analogous with traditional escrow arrangements.

c. **Omnibus and segregated custody accounts.** Crypto custodians typically offer one of two types of custody accounts: an omnibus arrangement where multiple clients' assets are commingled at a single blockchain address; and a segregated arrangement where each client has a dedicated blockchain address. In an omnibus arrangement, the custodian has undifferentiated on-chain control over the commingled assets, and individual client claims can only be resolved by reference to the custodian's internal records.

Advanced custody services

112. The custody models described above are ones in which a third-party provider offers services akin to 'safekeeping', or to those offered by an escrow agent. Readers may be familiar with securities custody services in traditional financial systems, whereby custodians may implement securities lending programmes by which securities under their control can be lent to third parties in return for payment. That helps generate an income stream for custodians (which custodians may or may not share with an end customer in exchange for consent to allow their assets to be lent in this way). Analogous services are available with digital asset custody, whereby a service provider with technical control over a digital asset can lend out that asset (or otherwise put it to work).

113. As introduced in Chapter 6, blockchain systems introduce the possibility of transferring control of digital assets to smart contracts, which may have features such as 'lending protocols', 'yield pools', or (in some specific situations) 'staking'. However, questions arise as to the uses to which an underlying asset might be put whilst under the control of that third party – e.g., if an asset is 'staked' by the operation of a smart contract (see 'What is validation?' in Chapter 3, above) or lent out, who is entitled to the return (akin to interest earned), and who carries the risk if such asset is not returned?

114. Addressing such matters involves interrogating the legal relationship between a user and third-party provider; the features of the technical implementation is unlikely to help resolve such issues. Once the relevant terminology and basic features of a custody arrangement are understood, the Panel anticipates that the resolution of most disputes which involve delegated control will not have significant technical, digital asset-specific nuances.

A short note on computer security

In the interest of simplifying exposition, throughout this Paper the Panel has implied that when a user digitally 'signs' a transaction or authorises an action be taken via a service provider's website or app, that user knows what it is they are approving. However, the reality is that vanishingly few human users can independently study a series of letters and digits purporting to be a blockchain transaction to satisfy him or herself that, if signed, the system will do what is expected. Instead, users rely on software (sometimes hardware in the form of physical wallets, often similar to a USB key) to construct the transaction, display it in human-readable form, and then compute and apply the signature. In short, when a user instructs their computer or mobile device to sign a signature, they are relying on that computer (and the software it runs) to have faithfully relayed what a particular transaction does, and to then approve that transaction as opposed to a different transaction.

The nature of such a system creates several opportunities for malicious parties to intervene and steal funds. If an attacker can compromise any system involved in the display or signing of a transaction, then that hacker can fool a user into believing they are approving one thing whilst actually approving something else.

At one level, such threats are conceptually the same as any threats faced by someone transacting online, and a full treatment of computer security is beyond the scope of this Paper. However, it is important to note that digital asset users and systems are often the targets of highly sophisticated and motivated adversaries. Many disputes involving digital assets feature situations where the technical evidence shows a particular transaction was signed for by a particular private key and that this key was possessed by a single person, yet that person denies having authorised the transaction. Litigants and decision-makers should be cognisant of the possibility that a user *did* sign a transaction (or cause it to be signed) without *realising* they were doing so because their computer or that of a service provider has been compromised. Forensic expertise is likely to be required to analyse such situations.

CHAPTER VIII. LAYERED CONTROL

115. Layered control is something readers will already be conceptually familiar with: traditional banking systems using payment rails for digital assets is an example of layered control. In such systems:

a. the customer controls the initiation of the payment instructions by interacting with the system which holds the authoritative record for the assets included in the payment (Base Layer or Layer 1);

b. the relevant bank's internal systems control the authorisation of the transaction, completing risk or fraud checks and verifying that the bank account has sufficient funds to effect the transaction (Layer 2);

c. a payment network such as SWIFT or Visa controls clearing and routing of the payments between institutions (Layer 3); and

d. a central bank or regulator maintains control through predefined and agreed-upon rules, and can intervene in extreme cases.

116. A similar structure of layered control can be seen with management of assets by custodial exchanges, discussed in Chapter 7. There, a customer of an exchange (who has passed the requisite KYC checks) will submit instructions to buy, sell, or make deposits or withdrawals (Layer 1). An exchange's automated wallets, risk engines, and compliance checks then control the decision on whether to approve or reject a particular request (Layer 2). Human operators or administrators may be in a position to intervene by freezing accounts or investigating complaints (Layer 3). Regulators and law enforcement agencies may then have an additional layer of control over the exchanges. Simple smart contract logic, discussed in Chapter 6, could also be considered a layered control structure. There, an end user interacts with a smart contract's functions, such as withdrawing an asset (Layer 1). The smart contract logic then automatically enforces programme rules (Layer 2). Regulatory oversight, if the system is operated by a regulated entity, may provide a further layer of control if intervention is necessary.

117. Architectures of layered systems will vary. In the event of a dispute involving some form of layered control, it will be necessary to consider each system on its own terms in order to analyse how control is implemented at each level. In this Chapter, the Panel

seeks to introduce common features of such structures so as to help provide a framework for navigating disputes which may involve analysis of layered systems of control.

118. Generally speaking, by way of a framework to help conceptualise such structures:

a. **Layer 1.** The starting point is the existence of a Layer 1 or a Base Layer, which can be described as the conceptual 'home' of the digital asset in question (usually being, in the context of this Paper, the blockchain). That asset must be programmatically 'locked' at Layer 1 before the asset can be bridged to another blockchain or before there can be a representation about, or 'wrapped' version, of that asset on another technical layer. In practice, in nearly all instances, an individual participating in services provided by another technical layer will directly hold the asset locked at Layer 1. Controls by other technical layers on how an asset is sent, deposited, or withdrawn, will still depend on Layer 1 verification.[30]

What is a 'wrapped asset'?

The high degree of 'connectedness' between technical layers facilitates the efficient movement of digital assets. This efficiency is, in part, enabled by the ability to represent a digital asset in a 'wrapped' form: an asset originating on one layer is represented by a token on another layer without creating a copy of or moving the original asset, but instead 'wrapping' the original asset. For example, when Alice's 10 ETH tokens on the native Ethereum blockchain are sent to an application or smart contract, a representation of those 10 ETH will be minted by the application to create 10 wETH (10 wrapped ETH). Alice can now easily transfer or move these 10 wETH around the new layer to, say, Bob with the original 10 ETH remaining locked on Ethereum. Only when Bob unwraps the 10 wETH, using an application, does the original 10 ETH unlock and get released to Bob. In summary, wrapping creates a tokenised representation of an asset.

What is 'bridging'?

Bridges exist between different blockchains to enable interoperability between the blockchains and so asset value on one blockchain can be represented on another blockchain. In a similar way to wrapping, assets are programmatically locked by a smart contract, or bridging contract, on the original blockchain and a corresponding asset is minted or liquidity

[30] Trustless smart contract logic will automatically enforce locking and unlocking based on someone's ability to prove ownership. Sidechains use the same concept of locking digital assets but they have independent consensus, block production and the security models compared to well known Layer 1s. Sidechain participants hold wrapped tokens in their wallet but redemption is controlled by operators of the side chain itself, not a smart contract on the Layer 1.

is released by the bridging contract on the other blockchain. For example, when Alice wants to use 10 ETH worth of value on the Polygon blockchain, she sends 10 ETH to the bridging contract on Polygon, where it gets locked. The Polygon bridging contract mints and sends 10 bridged ETH to Alice's Polygon address. When Alice sends the 10 bridged ETH to Bob, Bob now has the claim on the locked ETH. To reclaim the ETH, Bob sends the 10 bridged ETH to the bridging contract where they are burned, and 10 ETH are unlocked and sent to Bob's Ethereum address.

b. **Social recovery / guardian mechanisms.** As foreshadowed at paragraph 74 above, these typically act as an extra security layer. A social recovery wallet will involve its primary owner, as well as a set of guardians (which are typically trusted people, such as trusted institutions, hardware devices, DAOs, another wallet, or a dedicated guardian service). If a primary user loses his or her primary key, guardians can step in to approve a recovery transaction that sets a new key or otherwise restores control of the asset back to the original primary user. Typically they are instructed to take action by the primary owner of a wallet who has lost their primary key ("Hey Guardians, I've lost my key. I've initiated the recovery process. Can you log in and approve the recovery please"). The introduction of a guardian mechanism gives rise to the possibility that guardians might collude to unlawfully take control of an asset, but their level of power is generally restricted: they can only execute a recovery process and do not have spending authority, so they cannot move assets. With this said, the prevailing recommendation is that the owner of an asset should be the only person who is aware of who all of the guardians are, leaving the guardians unaware of one another's identity and therefore unable to collude.

c. The concepts introduced in Chapter 5 – e.g., multi-sig wallets – can combine operational security with the concept of recovery using guardians. The requirement for multiple co-signers to exercise control over an asset can be reconfigured by guardians in a recovery scenario – thus overriding a signer's control even while that signer retains his or her key. Recovery systems could also mathematically split a user's private key into multiple 'shares' which are distributed to guardians and, in a recovery scenario, might allow a majority of shares to be recombined to restore a lost private key. In such situations, the role of the guardians is limited to helping reconstruct the private key (they have no authority to control the movement of asset, although colluding guardians could in principle reconstruct a private key without

the user's consent). This approach of splitting a sensitive secret (such as a private key) into multiple fragments is a cryptographic method known as 'Shamir's secret sharing'.

d. **Administrative override / supervisory control.** As has been repeatedly noted, many systems will feature administrative override or a level of supervisory control which allows intervention in certain scenarios. Centralised custodial exchanges, such as Coinbase and Binance, under appropriate circumstances like fraud investigation, regulatory orders or sanctions compliance, have administrative control to freeze user accounts, blacklist users, reverse pending withdrawals internally, and enforce compliance holds. Note that such administrative override can be present even in decentralised systems, and can be enforced by smart contracts if they have been implemented in the relevant contract (which, once deployed, will execute the relevant override without discretion).

Finality time windows

119. Another way of conceptualising layered control is to consider the concept of finality, which is central to how layered blockchain systems work. There are two types of finality: (i) 'soft finality', where a transaction is accepted by a system other than Layer 1 and treated as complete on that basis; and (ii) 'hard finality', where a transaction is immutably confirmed on the Layer 1 blockchain. Generally, hard finality takes longer than soft finality; thus, digital asset projects will commonly operate on the basis of soft finality so as to enable assets to move freely within a system in order to provide a better user experience (that is, for the purpose of keeping the system moving, soft finality can be considered 'good enough'). However, until hard finality is achieved, the transaction in question can still be challenged or reversed.

A practical illustration

The Panel anticipates that the fact that there can be differing definitions of 'finality' for an action on a blockchain means that business transactions which incorporate on-chain and off-chain activity is likely to lead to disputes, meaning that readers may well be required to engage with the detailed operations of layered systems.

A worked example may help the reader conceptualise what the Panel is seeking to describe in this Chapter. Imagine you move $1000 from your digital asset wallet to your account,

held with a crypto exchange. The exchange can see the transaction has been executed through their web page; however, the underlying blockchain is yet to create the block which captures the transaction. During this interim period the finality is 'soft' and if the exchange believe you are trustworthy up to $1000, they are likely to effectively extend you credit and show $1000 in your account, which can be used for trading and so forth. Once the blockchain has 'caught up' and created the block which captures the transaction, the finality becomes 'hard', the exchange can see this by monitoring the blockchain, and the exchange is no longer extending you credit.

Throughout this process, you are none the wiser because the exchange web page obfuscates this complexity. In the context of layers, a Layer 2 application will use the same obfuscation: the application accepts an action executed within the application as 'good enough' to allow the user to continue uninterrupted, but the application will monitor the Layer 1 blockchain in the background so it knows when hard finality has been achieved.

On the rare occasion that the finality does not reach a hard state, then the exchange will determine how to react based on how it has been built. For example, the exchange could halt all further actions for the user and try to rollback their previous actions, or there may be manual intervention.

120. Finality time windows vary depending on the layered technology being used and whether there is a challenge period (that is, a period of time during which someone can contest the accuracy of data being committed to the blockchain). On one end of the spectrum, zero-knowledge proof systems (discussed further in Chapter 8, below) typically have the smallest finality windows, largely because the proofs involved are cryptographic and there is no challenge mechanism in place. On the other end of the spectrum, some systems (such as systems where transaction information is batched with other transaction information being submitted to Layer 1) can see multi-day windows in which someone can challenge the validity of the transaction information, and hard finality is achieved only upon the expiry of the challenge period. Between the two extremes lie encrypted systems, whereby transaction information is batched, encrypted, and submitted to Layer 1, but each submitted batch is programmatically verified to see whether a challenge needs to be made, with hard finality being achieved once verification is passed.

'Gas money'

At this stage, readers may be interested in the economics of layered control systems. Layered systems commonly share standard economic fundamentals, though there remains a wide variety of fee models and pricing mechanisms. Although the detailed economics of the digital asset system is beyond the scope of this Paper, it may be useful to understand the basic concepts to the extent transaction fees feature in disputes.

The concept of 'gas' is typically used in layered systems and is the unit of operational cost on many blockchains. It was originally devised by the architects of Ethereum. Every operation on the blockchain – such as sending a digital asset, storing data, or making a query – consumes a fixed amount of 'gas'. The purpose of gas is to prevent abuse of the blockchain (for example, intentionally coding infinite loops) and to create a free market where scarce space on the blockchain is allocated to users who are willing to pay. Gas has a price, and the user must pay a fee for the amount of gas used to the operators of the nodes which form a blockchain network.

Fees on Ethereum are algorithmically set at a protocol level. Other blockchains operate very differently to Ethereum. For instance, Bitcoin is fully decentralised and users attach fees to their transactions, with miners prioritising higher-fee transactions, thereby creating a fee market based purely on supply and demand. These parameters are fixed in the bitcoin blockchain protocol.

Solana is another example which operates differently. Solana is a Layer 1 blockchain with transaction fees algorithmically controlled based on computing and storage demand. Fees are paid in the native token of the blockchain ('SOL'), which has a transparent predefined issuance model. The fees model is built into the on-chain Solana governance program, and SOL token holders have the right to vote on protocol parameters (including on the fees model) – the parameters are therefore influenced by Solana Labs (the developer) via protocol upgrades, and controlled by the wider community based on voting consensus.

Generally speaking, more complex transactions command higher fees. For example, Alice sending 1 USDC to Bob uses very little gas because the computation and blockchain storage required is very low. By contrast, minting a new NFT is very costly because it requires the creation of new storage on the blockchain, which uses lots of gas. Storage is the most gas intensive aspect of a blockchain because it means creating a data record which, in theory, lasts forever (and forever is expensive!).

For present purposes, the Panel anticipates that it suffices to understand that most blockchains adopt one or two or a mix of models: the economic parameters of a system are either controlled by the original developers, or subject to governance voting by the community. The former is more common, and the latter tends to be less well-developed.

The future of layered systems

121. The reader may or may not be familiar with the advent of quantum computing: that is an area of computing which leverages principles of quantum mechanics to solve complex problems significantly faster than classical computers. Some of the cryptographic techniques which secure blockchains today will very likely be compromised in due course once quantum computing becomes a reality. There currently exist a very limited number of quantum-resilient systems, though the vast majority of systems are making plans for new technical upgrades and governance models for a post-quantum world.

122. At time of writing, most layered systems are taking a similar approach to achieve quantum resilience, by aligning to the U.S. NIST's post quantum cryptography standards. Layer 1 and Layer 2 systems will likely incorporate post-quantum key types and protocol options, then encourage users to migrate to these new keys (e.g., by prompting existing users to add new post-quantum keys), and later see blanket implementation once most users have upgraded. Smart contracts and protocols will also likely incorporate post-quantum verification through community-approved migrations.

123. All of this means that, beyond the basic structure and concepts being described, much of what is discussed in this chapter may be subject to change in the near-future. It is difficult to predict how quickly post-quantum implementations may take place, as this will depend on the publication of standards, security urgency and demands, and the speed of implementation at lower layers (because a Layer 2 system cannot be full quantum-resilient if its Layer 1 system still depends on current cryptography). The NIST has finalised its core post-quantum algorithms, so at time of writing, developers are already able to consider the engineering choices that are available to change their systems. The Panel estimates that full migration will be completed by 2030-2035.

CHAPTER IX. PRIVACY-ORIENTED CONTROL

124. Many blockchain systems (e.g., Bitcoin or Ethereum) are public and transparent. This means that all transactions and wallet balances are visible to anyone. That can create challenges where there is a demand for privacy. To address this, there are emerging technologies which focus on ensuring privacy such that transactions on the blockchain cannot be identified or traced back to any user.

125. In the Panel's view, these technologies could introduce complications when analysing control of assets. In almost all privacy-enhancing methods, whatever mechanisms are used, the 'true' owner will still be able to cryptographically prove his or her ownership. However, in the absence of cooperation from the true owner, third parties (including the Court) will encounter significant difficulty in resolving disputes about disputed ownership and control. This is likely to be exacerbated where privacy mechanisms are deployed.

126. In this Chapter, the Panel explains some of the emerging technologies and mechanism available, which could be relevant and necessary to understand and resolve disputes.

The emerging technologies

127. To the Panel's knowledge, there are over 750 privacy-related projects in the market. However, it is possible to identify four 'mainstream' encryption technologies on which these projects rely, as follows:

a. **Zero-Knowledge Proofs ('ZKPs')**. These technologies help ensure that valid payments can be verified without exposing sensitive transaction details. ZKP-based privacy protocols will hide transaction amounts, asset balances, and relationships between inputs and outputs. The public keys of both senders and receivers can remain hidden (this being a hallmark of privacy-preserving coins like Zcash and other systems featuring shielded addresses). ZKP technology can prove membership of a set without revealing which member is which and specific identity is private inside a group. So, rather than hiding the entire transaction as with shielded transactions, inclusion within an approved set can be proven by an individual. Similarly, ZKP technology enables selective disclosure of facts, for

example, “I am over 18” or “I am not on a sanctions list” replaces full identity and KYC disclosure to achieve regulatory approval.

b. **Fully Homomorphic Encryption (‘FHE’).** These allow computations to be performed on encrypted data, meaning that data remains encrypted at all times even while transactions, balance checks, and smart contracts are executed. Those performing the computation (that is, blockchain validators and smart contract code, explained at Chapter 3 above) never see the underlying data, which ensures that no entity other than the data owner can view the data in decrypted form. Decryption is only possible by the holder of the relevant private key (but with user consent, decryption keys could be provided to others). For the moment, FHE technology is nascent, but it is expected to develop to have wider applications, e.g., in confidential auctions or computation over healthcare records.

c. **Mixers**. This refers to non-custodial mechanisms for achieving privacy by intentionally introducing ambiguity, for instance by pooling funds together, using partial cryptographic proof rather than full data encryption. It also introduces ‘relayers’ in the withdrawal process, which are service providers that can pay transaction costs on users’ behalf, to ensure that there is no transaction history connecting to the transaction participants, and further obfuscate the relationship between those parties. The deposit-withdraw linkage is broken thereby achieving anonymity.

d. **Trusted Execution Environments (‘TEEs’)**. TEEs refer to hardware-based encryption technologies – i.e., a secure and secure area of a hardware processor. These can support more complex use cases than ZKP, FHE, and mixers (currently) can, and they use technologies that are considered more tried-and-tested and mature. An example is Intel’s SGX TEE used by TEN Protocol, which has been available since 2015: the TEN Protocol uses TEEs so that transactions are submitted to a network encrypted with a cryptographic key known only to that network; responses, event log queries, or transaction outcomes are encrypted by the network using a ‘viewing key’, known only to the transaction’s participants. The computation and state storage are hidden from everyone, including node operators, save for those with the necessary keys.

128. Alternatively, another way of understanding and conceptualising this area may be to consider what the technology in question aims to do, rather than the technologies themselves. In this regard, privacy-oriented mechanisms could be said to fall broadly into two categories:

a. **Concealment technology**. First, there is a category of technologies which simply hides the identity or the key of the participant controlling an asset. The effect is that even though outside observers cannot see who controls an asset, the actual private key owner still retains full control. Typically, 'confidential transactions' and 'stealth addresses' refer to concealment. ZKP is an example of concealment technology, in that the protocol hides transaction details completely (including sender, receiver, and amounts). However, the controller of an asset is still uniquely defined by possession and control of a secret key, and only the key holder can spend or prove control to others by cryptographic means.

b. **Ambiguity technology**. By contrast, ambiguity technologies are conceptually different, because they are designed to create uncertainty over which party among a group of participants is in control. The ambiguity lies not in who controls the asset after a transfer is made, but in tracing which input belongs to which participant (i.e., who among a group effected the transfer), with the result that multiple parties could appear as plausible controllers of a given asset. Mixers and Ring Signatures are common examples. With Mixers, asset addresses are mixed such that after the transaction, the link between incoming and outgoing addresses is obfuscated and several participants could feasibly be associated with a particular asset. With Ring Signatures, a transaction is signed by a "ring" of possible signers, but only one of them is actually controlling the asset. All members of the ring are equally plausible as controllers to outside observers.[31] In both examples, there is ambiguity about which key or participant controls an asset, until such time as an

[31] For a worked example: imagine there are 5 people in the ring, and one person signs the transaction. The one signatory does so using his or her own private key, plus the public keys of the other 4 people to form a ring. The signature proves one of the private keys corresponding to these 5 public keys signed this message, but not which one. There is therefore one real signer and five plausible signers. In a properly implemented ring signature scheme there is no cryptographic way to determine which of the 5 keys signer (i.e. no "master key that can later reveal the signer). This is different from encryption (where data might be decrypted later) because ring signatures are information-theoretically ambiguous once created.

asset may actually be spent or disposed of, and the participant in control thereby reveals him/herself.

129. A further distinction that could be drawn as between different privacy mechanisms is the extent to which there may exist an intermediary or custodian (as explained in Chapter 7 on 'Delegated Control'). On this, there is significant variation as between different technologies: some are explicitly designed to be trustless and decentralised, others less so. It is difficult to make generalisations as between technologies, but in broad terms:

a. Protocols such as ZKP protocols, Ring Signature systems and confidential transaction protocols are trustless and non-custodial, meaning there is a lack of a centralised operator.

b. By contrast, ambiguity technologies, like Mixers, can require an intermediary or custodial step where the mixer service is operated by a third party acting as 'the mixer': asset owners send their assets to the mixer, which takes temporary custody of the asset, and then returns the assets after it has been 'mixed' (and thus anonymised).

c. However, not all Mixers operate in the same way. There also exist decentralised or non-custodial Mixers, which rely on smart contracts or cryptographic protocols to automate the 'mixing': Tornado Cash is a well-known decentralised mixer.

130. Pausing here, to take the analysis further, it is necessary to relax one of the assumptions discussed variously above, by which we assumed that a centralised operator always has administrator-level authority to halt or override transactions and users' instructions (e.g., as a bank operating a banking system would):

a. Here, the existence of a centralised operator in some of these technologies does not necessarily mean they have the capability to 'override', or (for instance) could be ordered to reveal transaction or controller details. Privacy projects in general do not provide for a centralised intermediary who acts as administrator (although some may).

b. Conversely, just because a protocol is decentralised and non-custodial, that does not mean there may not sometimes be a pause or 'kill switch' functionality, which allows a contract administrator to halt the protocol in case of emergencies (e.g., a hack). In the case of Arbitrum, for example, there is a 'Security Council' which is empowered to make decisions on emergency action, meaning that although it is a decentralised and trustless protocol, there are nevertheless some with administrator-level control.

c. A further alternative is that there are some protocols which can be deployed in 'safe mode', meaning that the network can only operate when a number of known group participants publish their public cryptographic key and receive their segment of the encrypted master seed in return. This continues until the required threshold of master key segments is met and the network becomes operational. In such technologies, an owner's control is not tempered by the existence of an administrator with the authority to override such control, but by the need for cooperative action by other participants.

131. The absence (more often than not) of a centralised custodian raises the question of whether and to what extent a Court, regulatory body, or other decision-maker could ever compel disclosure of information that is protected by privacy mechanisms. Like any other technological solution, privacy-oriented systems could be designed to provide disclosure when required (e.g., for auditing or regulatory purposes). However, to the Panel's knowledge, compelling disclosure without an asset owner's cooperation is largely impossible from a technological perspective. This is because, generally speaking, developers do not possess keys or have any other mechanisms to forcibly reveal encrypted transactions. But there are some exceptions:

a. There are some privacy protocols which might integrate selective disclosure mechanisms, for instance by offering an optional or mandated transparency key which could be accessed by regulators (such as Aztec's enterprise solution and TEN Protocol, which are designed explicitly for regulatory compliance scenarios with developers allowing third parties to view some or all information).

b. Time-based access is another approach. TEN Protocol, for example, provides application developers with the ability to define a time delay after a transaction is submitted, at which point all the transaction details can be decrypted by anyone.

For example: a protocol might be programmed such that "All transactions over £1000 become visible after 1 month".

Practical consequences for resolving disputes

132. The unifying theme of the forms of control discussed so far is the fundamental principle of blockchain technology that every transaction is cryptographically signed with private keys. This helps create immutable evidence that any given transaction has been authorised by the relevant asset controller. As has been explained, it is a feature of blockchain-based systems that, in general, it can be very difficult to prove a party's control of an asset or participation in a transaction in the face of their denial or non-cooperation. Nevertheless, the use of cryptographic digital signatures in the form of a private key means that there is still some mechanism for uniquely identifying a controller and ensuring transactions cannot be forged without detection, even if the individual person or entity attached to that private key cannot necessarily be identified. Similarly, in instances of multi-party control, such as multi-signature wallets where smart contracts enforce joint authorisations, a clear audit trail of which keys participated can be provided if the protocol developer makes that information available, and that may assist in identifying a controller even if the individual identity attached to that private key cannot necessarily be identified based on this data alone.

133. The technologies discussed in this chapter undermine that framework, either because the details necessary to inform such an audit trail are concealed or deliberately made ambiguous. Thus, while the blockchain still serves as an immutable transaction log regardless of the privacy system in use, in practice it may be that the log cannot be read – e.g., because transactions are submitted in the form of an encrypted rollup to the blockchain (in the case of TEN Protocol), or they can cryptographically sign for a transaction without revealing the participant (Ring Signatures and ZKPs), or they mix coins to obscure their provenance (Mixers).

134. The effect of these technologies may be significant in dispute resolution. A litigant or Court analysing the relevant logs, signatures, and other evidence of transactions can confirm that a transaction was duly signed and authorised by a valid key, but these technologies often render it impossible to identify any characteristics about the

controller. Indeed, a controller may have means of positive deniability through use of these mechanisms.