
Smart Contracts: Practical Tips from the Law Commission's Advice to Government

By Jack Castle¹

The Law Commission has published an Advice to Government on 'Smart Legal Contracts'. It examines the preparedness of English contract law for technological innovation, including the use of code as a contractual language and the impact of blockchain. What are the benefits and pitfalls of contracts involving these technologies? Where, how, and can they be enforced?

Scope of the Advice

1. The Law Commission's Advice discusses 'Smart Legal Contracts', which it defines as "*a legally binding contract in which some or all of the contractual obligations are defined in and/or performed automatically by a computer program*". It looks at contracts formed by, or on behalf of, contracting parties with some involvement of a computer program or code. This includes but is not limited to smart contracts on a blockchain or other distributed ledger technology ("DLT").

The Technology

2. Some of what the Law Commission describes as Smart Legal Contracts are already in use. Online shopping and internet banking, for example, use a level of

¹ The author would like to thank Peter Susman QC and Henry Warwick QC for their comments on an earlier draft of this alerter.

automation to form and/or fulfil the contractual obligations of one or more parties, as do algorithmic trading strategies.

3. The Law Commission however seeks to look further than the present. One of the technologies discussed are ‘smart contracts’. These are programs which run automatically without need for human intervention when certain conditions are met. However, they are not necessarily ‘contracts’ in the legal sense. Some smart contracts are not Smart Legal Contracts (as defined by the Law Commission) – they are just programs.
4. The Law Commission also discusses DLTs. DLT is a digital ledger which can only be added to. Once data is added to the ledger it cannot be modified or erased. Data is added to the ledger in ‘blocks’. Identical copies of this ledger are hosted simultaneously on multiple devices in a network (called ‘nodes’), with no one copy being authoritative (hence ‘distributed’). As well as simple data, certain blockchains are able to have smart contracts added to them as part of a new block.
5. When a block is to be added to the ledger, each node constructs its own copy of the newly enlarged ledger. Then the nodes decide which new copy is correct using a consensus mechanism. When the nodes have decided what the correct version of the new ledger should be, all nodes update themselves with the new, correct copy of the ledger. The goal is to create an authoritative record without the need for a centralised authority to administer that record.²
6. The type of consensus mechanism used depends on the blockchain. The most well-known public blockchains, including Bitcoin and Ethereum, use a system where each node competes to solve a difficult mathematical puzzle. The prize is to choose

² A recent judicial description of the technology can be found in *Tulip Trading Limited v Bitcoin Association for BSV & Oths* [2022] EWHC 667 (Ch) at [16]-[19].

which block of data to add to the chain, and receipt of an amount of the blockchain's native cryptocurrency. This is how new cryptocurrency is created, and the process is called mining. It is this process that gives rise to Bitcoin's famously large energy usage. This cryptocurrency can be spent on goods or services where it is accepted, or exchanged for another currency: either a different cryptocurrency or a national currency, using cryptocurrency exchanges.

7. When smart contracts are combined with DLT, proponents say that transactions could be executed and recorded indelibly without need for a centralised database, enabling (for example) the provenance of goods to be monitored from hand to hand, tracking food from production to consumer, confirming the authenticity of artworks, or providing a definitive register of ownership.
8. There is also scope for two programs to interact with each other without the need for human intervention. For example, two parties could each deploy computer programs on a cryptocurrency exchange platform, with the programs set to place orders to buy and sell cryptocurrency on the platform at algorithmically determined prices. Subsequently, one party's program could place an offer to sell cryptocurrency, and the other party's program would automatically accept that offer, leading to an exchange of cryptocurrency between parties. These are the facts in [Quoine Pte Ltd v B2C2 Ltd \[2020\] SGCA\(I\) 02](#), discussed below.
9. In general, the Law Commission found the common law sufficiently ready, or at least flexible enough, to deal with issues raised by Smart Legal Contracts. It did, however, raise a few complexities that practitioners should note.

Making Smart Contracts Legal

10. The first issue is whether a specific program or its functioning meets the specifications for the common law to recognise it as a contract. When is a smart contract a Smart Legal Contract? That is, does an interaction with a given program

constitute an offer, or an acceptance? Is the bargain for consideration, with the intention to create legal relations? If any formalities requirements apply to the contract, does the program satisfy them?

11. A common scenario might be as follows. A party codes a program that will cause a product to be dispatched to a customer on payment of a certain amount. This would be an offer to sell the product, accepted by a buyer paying that amount. Consideration would be the promise to exchange the product and money. Performance would then be carried out automatically by the program ordering the product to be dispatched. So, aside from the automated nature of offer and performance, there may be little difference between a Smart Legal Contract and any other contract.
12. These matters are unlikely to cause significant problems in practice, and indeed have not so far. The Law Commission's Advice does however raise a number of important points of practice specific to Smart Legal Contracts where novel technology is used.
13. First, programs on public blockchains lend themselves to a unilateral contract analysis. If a program is added to the Ethereum blockchain that (for example) offers a token of some kind in exchange for an amount of Ether, the offer will be to everyone who can access the blockchain. Placing the program on the blockchain will likely be an objective intention to make such an offer, and anyone will be able to accept it. Practitioners will need to take care to specify any limitations on who can accept and how.
14. Second, agreements between pseudonymous parties are likely binding. Contracts formed on an anonymous blockchain can therefore have legal effect. Clearly, however, enforcement against a pseudonymous counterparty will be challenging. Care needs to be taken in choice of counterparty, or for the contract to provide for some form of identity check.

15. Third, there is support from the Singapore Court of Appeal in *Quoine Pte Ltd v B2C2 Ltd* that offer and acceptance can be made between computer programs on behalf of their counterparties, automatically, without explicit human input. An offer made by one algorithm can be accepted by another, binding the owner of each. This is a potential gift to business efficiency, and a potential curse in terms of the liability that might result.
16. As to formalities, the Law Commission considered that the definition of ‘writing’ in the Interpretation Act 1978 was apt to cover high-level computer code.³ As such, though context-specific, a Smart Legal Contract may be able to constitute signed writing, thus fulfil certain formalities requirements. The exception is deeds, which are required to be witnessed and for that witness to attest to the signature. The Law Commission doubts whether “smart deeds” will be effective as deeds.

Interpretation of Smart Legal Contracts

17. It may be thought that Smart Legal Contracts would not need interpretation: the code does what it does with no room for ambiguity. Unfortunately, as readers may know, there is sometimes a difference between what a computer program should do and what it does – whether because of malfunction, or an inaccurate description of the code’s function in natural language, or an error in the code itself.
18. For example, where a Smart Legal Contract is half in natural language and half code, a Court may have to consider both when construing the contract as a whole, including deciding which to give priority. Or a coded contract may not perform as

³ “High level” code, or source code, is code entered and understood by human beings consisting of words and symbols. This is in contrast to ‘low level’ code, generally known as machine code, which is understood by computer hardware and is typically in binary form.

one or both parties expected, whether due to an unexpected outcome of correct code, or bugs in the code that were not picked up. These would be ambiguities in the contract itself, requiring interpretation. The Law Commission considered the law of interpretation should not be disappplied for Smart Legal Contracts.

19. Contracts are construed by reference to the ordinary meanings of the words used as they would objectively be understood by persons in the position of the parties. One obvious difficulty with contractual interpretation of code is that code is not meant to be read by ‘persons’. The question therefore becomes: who is the notional objective arbiter of the meaning of a program?
20. The Law Commission saw two choices: the code should be interpreted as it would be by either (i) a functioning computer, or (ii) a person with knowledge and understanding of code. It opted for the ‘reasonable coder’ test, noting that expert evidence may assist the court to ascertain what the reasonable coder would have understood.
21. However, if the parties have agreed that the contractual terms should accrue in the machine or other low-level code – interpretable only by machines – then the reasonable coder can have no insight and the code may mean whatever it does when executed. That the true agreement is in low level code may prove to be a fruitful argument for defendants, providing an argument that performance rendered accords with the contract; a contract which permits of no interpretation whatsoever. With low-level code, what you receive is precisely what you contracted for.
22. Although not disapplying the law on implication of terms, in practice implication of terms into a Smart Legal Contract may have limited scope if the express terms are in code rather than English. If certain functionality is wanted it will have to be coded for: a computer can make no assumptions, and the Law Commission notes

that the behaviour of the code is likely to be a strong indicator that the agreement is coherent and complete.

23. The Law Commission also notes that one of the reasons for automated performance is that elements of trust and cooperation are removed or minimised, perhaps making an implied term of necessary cooperation more difficult to argue for or less relevant. However, of interest is the Law Commission's opinion that a natural language term could be implied into a contract wholly or partially in code. An implied term that a party not take steps to prevent performance may be a good example.

Remedies Available

24. A key benefit of Smart Legal Contracts is automated performance. Non-performance may arise infrequently in practice, though this will be industry-dependent. Naturally, tangible goods may still arrive late or damaged even if the legal contract is smart.
25. The remedies available in contract are also applicable to Smart Legal Contracts.
26. The Law Commission thinks that rectification may take on new prominence. There is clearly ample opportunity for code in a Smart Legal Contract to function other than in a way that implements the actual intention of the parties.
27. Similarly, situations where code is involved may give more frequent rise to common mistake and unilateral mistake, though the Law Commission did not recommend that the concepts themselves should be broadened.
28. That said, where programs accept offers made by other programs, there may be a high bar to proving mistake. Drawing on *Quoine*, the Law Commission's discussion of this issue as related to unilateral mistake is instructive (at [5.55]-

[5.59] and [5.67]-[5.76]). In *Quoine*, the majority of the Singapore Court of Appeal framed the question in terms of the intention of the coder (at [103]):

when programming the algorithm, was the programmer doing so with actual or constructive knowledge of the fact that the relevant offer would only ever be accepted by a party operating under a mistake, and was the programmer acting to take advantage of such a mistake?

29. The Law Commission appears to recommend that judges should follow the majority in *Quoine*, placing the requirement for knowledge of mistake on the programmer, and extending the scope of unilateral mistake to require only constructive knowledge that a counterparty would be mistaken as to a term of the contract. This would still be a broadening of the concept in English law, albeit in a specific factual situation.
30. The doctrine of frustration may also have greater relevance. Smart Legal Contracts present a new range of potential factors outside the parties' control that render performance impossible or radically different to what was agreed.

Remedies: Practical Considerations

31. Although generally optimistic in its views that Smart Legal Contracts can be accommodated within current contract law, the Law Commission does acknowledge that automation may limit the remedies practically available.
32. Problems with rescission arise if a smart contract is on DLT. DLT is, by design, write-only. Actual rescission is impossible on blockchain. In its discussion of this issue (at [5.96]-[5.103]) the two options proposed are that (i) the Court order 'equal and opposite' transactions to be entered into on the blockchain, or (ii) the Court value the exchanged benefits in money and order restitution of the value off-chain. But this is not 'unwinding' the transaction.

33. More acute problems occur with any remedy that stops or ceases performance of the contract: injunctions (both interim and final), freezing orders, and any form of termination. If a contract is being performed, any power the Court or a party has to stop it will be subject to the ability of the program to be stopped. This in turn depends on whether the program has been coded with the functionality to be able to be stopped. If it cannot be, the remedy will be impossible to obtain – Smart Legal Contracts cannot obey court orders unless they are given that functionality.
34. It is sensible therefore that all parties insist on code that allows the program to be arrested once begun, to maximise the remedies available. The lack of actual rescission on DLT is another practical consideration when deciding on the specifications of the Smart Legal Contract to be used.

Consumer Law

35. Many consumer rights rely on the notion of the ‘average consumer’, and this average consumer is unlikely to be able to read and understand code.
36. As such, businesses entering Smart Legal Contracts with consumers need to explain the function of their code in appropriate natural language, or risk their terms failing the transparency test in s.68(1) Consumer Rights Act 2015 (“**CRA**”). Failure to do so may also make those terms ‘unfair’ under s.62 CRA, or a misleading action or omission under the Consumer Protection from Unfair Trading Regulations 2008/1277.
37. Consumers have various rights to treat the contract as being at an end or to vary it. Consumer-to-business Smart Legal Contracts therefore need to be coded to be able to be terminated or varied to facilitate these rights.
38. From a data compliance viewpoint, businesses need to be wary of publishing personal data to a DLT, where it can be accessed by anyone and cannot be erased.

Jurisdiction and Choice of Law

39. The decentralised nature of DLT causes obvious problems ascertaining where a contract was formed or executed. As the Law Commission notes, that is the point of DLT. As such, this section (Chapter 7) of the Law Commission’s Advice is technical, somewhat speculative, fascinating, and beyond the scope of this alert. Indeed it is somewhat beyond the scope of the Advice; a further project on conflict of laws in the context of emerging technology is planned.
40. On jurisdiction, there is no clarity other than to say it will be fact dependent (the difficulty of the Law Commission’s job further exacerbated by the call for responses being issued in December 2020, a time of considerable uncertainty about which private international law rules may pertain). “Where the contract is performed” could be the location of either of the parties, the location of the server, or (for DLT) the location of some quantity of the participating nodes.
41. An interesting method of establishing jurisdiction may be to regard the relevant coder as an agent of one or more parties to the smart contract (at [7.42]). The idea that computer programs could themselves be agents was dismissed (at [7.46]) because an agent has to have a “mind”.
42. On the subject of applicable law, the Law Commission considers Rome I to apply to Smart Legal Contracts. Two major issues were identified:
- a. The location of digital assets;
 - b. Determining the location of particular actions, such as the place of performance or breach when the actions “take place” on a distributed ledger or other decentralised system.
43. As to (a), there are now a few first instance interim decisions holding that the place where the asset’s owner is domiciled is a strong indicator of location (*Ion Science Ltd v Persons Unknown* (unreported) (21 December 2020) at [13] and

Fetch.AI Ltd v Persons Unknown [2021] EWHC 2254 (Comm) at [14], considered in *Tulip Trading* [2022] EWHC 667 (Ch) at [139]-[158]). But, as the Law Commission notes, HMRC's view is that an exchange token (a cryptoasset that is native to a cryptoasset exchange) is located in the place where its beneficial owner is resident.

44. As to (b), it seems unlikely that a court would consider the location of nodes hosting the Smart Legal Contract to be relevant, because that may be entirely arbitrary. But a number of factors could be relevant, including the identities and locations of the parties, location of any real-world performance, the location of the creator of the program or asset, or the location of any private key.
45. As ever, best practice is to agree jurisdiction and choice of law in advance. The Law Commission recommend this be done in natural language, not within the code itself. It also notes that 'Law' for the purpose of Rome I is the law of a country, so choice of a particular platform's protocol would not be a choice of "Law".

Conclusion

46. Although English law is ready and able to facilitate Smart Legal Contracts, practitioners will need to consider in advance the ramifications of using code to set out legal obligations and the automated nature of contractual performance.

Jack Castle

1 April 2022