Computer Law & Security Review 55 (2024) 106035

Contents lists available at ScienceDirect

Computer Law & Security Review: The International Journal of Technology Law and Practice

journal homepage: www.elsevier.com/locate/clsr

From smart legal contracts to contracts on blockchain: An empirical investigation

Fabio Bassan [a,*], Maddalena Rabitti [b]

[a] Professor of International Law, Roma Tre University, Italy
[b] Professor of Economic Law, Roma Tre University, Italy

ARTICLE INFO

Keywords:
Blockchain
Smart contracts
Smart legal contracts
Contracts on chain
Regulation

ABSTRACT

The issue surrounding the nature and function of smart contracts in the context of legal relationships has garnered significant attention from the European and national legislators, regulatory bodies and legal scholarship. Sections I and II of this essay give an account of the results of the ongoing doctrinal debate, which is not univocal. The objective is to provide an assessment of both the advantages and limitations associated with smart legal contracts. In Section III, the authors introduce a novel negotiation process termed "contracts on chain". This process enables parties to engage in negotiations, formalize agreements and execute contracts directly on the blockchain. Consequently, this negotiation approach serves as a potential bridge between the realms of Web 2 and Web 3. Further, it offers a user experience akin to online contracts but benefits from the inherent capabilities of third-generation blockchains. Albeit on-chain contracts can be deployed on both private and public blockchains, the authors express a preference for their use on the public blockchain within a "logical platform". This choice allows to enhance regulatory compliance and mitigate the effects of decentralization on liability regimes, while simultaneously optimizing the efficiency gains of public blockchains. Notably, this approach ensures a level of protection commensurate with that offered by private blockchains. The ultimate goal of this innovative process is to streamline the ongoing technological transition and cultivate greater trust within the market for emerging technologies.

Introduction

The shift from Web2, representative of the current digital platform paradigm, to Web3, characterized by distributed ledger technology (DLT) and blockchain, represents a transformative transition. Similar to any period of transition, this shift requires the implementation of value-driven decisions to mitigate potential adverse consequences in the long run.[1]

Thus, it becomes clear that, despite the prevailing slogans extolling the virtues of absolute disintermediation and decentralization as ideal forms of governance,[2] on one side, and a reluctance to regulate as driven by the fear of stifling digital innovation, on the other side, there is an urgent need for negotiated rules and tools to proactively address or reduce potential risks that would otherwise pose unmanageable challenges.

Within the realm of blockchain technology, which serves as the foundational infrastructure enabling the development of products and services via smart contracts, divergent doctrinal perspectives come to the fore. These viewpoints frequently manifest a pronounced ideological character. Some scholars advocate a fully decentralized structuring of activities, which blockchain supports. By contrast, others rule out the possibility of using smart contracts to manage (the legal) relationships between two or more parties. To date, the two positions seem hard to reconcile: the first aims to bridge the gap between current and future

* Corresponding author.
E-mail address: fabio.bassan@uniroma3.it (F. Bassan).

[1] For a brief illustration of the elements characterizing this transition, see: F. Bassan, Web 3 in Transition, *in* CPI-Tech Chronicle, *2023.*

[2] See: A. Wright - P. De Filippi, *Decentralized Blockchain Technology and the Rise of Lex Cryptographia*, SSRN Electronic Journal, 2015, pp. 15-17; G. Paquet - C. Wilson, *Governance failure and the avatars of the antigovernment phenomena*, CoG Working Paper, 2015, pp. 16-27. From a more dialectical perspective, see: M. Atzori, *Blockchain Technology and Decentralized Governance: Is the State Still Necessary?*, Journal of Governance and Regulation, Vol. 6, Issue 1, 2017, pp. 15 - 21 e 25 – 32; D. Yermack, *Corporate governance and blockchains*, Review of Finance, 2017, pp. 7.31; B. Arruñada - L. Garicano, *Blockchain: the birth of decentralized governance*, Economics Working Paper Series (WP n. 1608), Universitat Pompeu Fabra, 2018.

https://doi.org/10.1016/j.clsr.2024.106035

Available online 5 September 2024

developments and move toward full decentralization, while the second rejects the need for a bridge given the limits of the blockchain environment. Among these, the most relevant is the risk that attempts to navigate between – i.e. translate – programming code, machine language, and natural language are futile and will produce insurmountable confusion. Accordingly, in the context of (legal) contracts written in natural language, it is hard to promote the circulation of smart legal contracts on a blockchain ecosystem absent 'certified' translations.

We contend that both approaches risk shortcomings by overlooking the potential value of a transitional link (a bridge) between the state of the art and the future development of blockchain. This bridge can effectively exploit blockchain features to enhance contractual certainty and security, permitting the partial self-execution of contracts and providing efficient mechanisms for dispute resolution. In this article, we introduce a concept of a "bridge" that originates from current online contracts and that leverages the recent advancements in blockchain and smart contracts.

This evolution, evident in some contemporary blockchains (which we will refer to as 'third-generation' blockchain), transforms the blockchain landscape into a robust, open-source infrastructure, purposefully designed for adaptation and flexible enough to facilitate a harmonious connection between the blockchain ecosystem and real-world scenarios, an infrastructure that bridges the gap between the language expressed in lines of code and natural language. This technological advancement holds significant theoretical and practical significance as it combines the benefits of digitalized negotiation processes on blockchain with the functionalities of smart contracts. Via smart contracts, the parties to a contractual operation can engage in transparent negotiation and efficient execution. The idea is to create a new negotiation process, which we define as "contracts-on-chain". As an expression of the union between technology, legal expertise, and legal design, this concept can support the ongoing transition while at the same time acknowledging the aspiration of the European Union to foster innovation with "reliable" smart contracts.[3]

At the outset, we bound the scope of the investigation in three dimensions. Firstly, we adopt a "European" perspective, one which operates under the regulatory framework of the civil law system, as opposed to the common law system. This perspective benefits from a path outlined by European institutions that, although not yet finalized, is clearly defined in terms of its direction and objectives.[4] Secondly, our analysis is confined to the financial sector and Fintech applications, given that this is thus far the most developed realm that is subject to considerable attention from European legislators and regulators. Moreover, the body of international literature on this topic is already significant. Thirdly, and as far as technology is concerned, we have decided to operate on a *third-generation* public blockchain,[5] whose technical features allow for the use of new tools to build contracts-on-chain.

In the realm of public blockchain, we conceptualize 'logical platforms'. These platforms, analogous to what currently exists in cloud computing, regulate access and activities conducted on the platform.[6] This technological model increases the potential of public blockchains while concurrently safeguarding the control typical of private blockchains.[7] This is a model that, within the current state of technological advancement, better allows the development of Contracts-on-chain.[8] This model averts the shift towards decentralization without oversight, as it is with the Decentralized Autonomous Organization (DAO), because decentralization characterizes the underlying blockchain and not the logical platform operating atop it. Hence, our research has both theoretical and practical implications and marks a shift from decentralized finance (De-Fi) to distributed finance (Di-Fi), meaning a finance operating on logical platforms coordinating each other (hence, distributed) and using decentralized blockchains. This entails the presence of various logical platforms operating on one or more public blockchains interconnected according to a distributed model.

The need to ensure compliance with a minimum level of protection (necessary and sufficient), i.e. to guarantee users of financial services and contracts on blockchain an equivalent level of protection to that which they have in the real world, has also been recognized by Italian institutions. The Bank of Italy (National Central Bank) has initiated a process to identify guidelines for smart contracts in the financial sector. There is a need to establish rules also in terms of contractual safeguards to ensure a complete balance between the interests at stake, innovation (including in the financial field), market trust, and user protection.

This work fits into this context in order to outline a brand new negotiation mechanism. The aim is to enable – on blockchain – the

[3] The notion of reliability in smart contracts is recurring in the regulatory framework established by the European legislature. Specifically, in Regulation (EU) 2022/858, which pertains to a pilot regime for market infrastructures based on distributed ledger technology (DLT Pilot Regime), the legislature acknowledges that practice reveals that 'smart contract' protocols have not yet been subject to any transparency, reliability, or security requirements (see Recital no. 5). The Regulation emphasizes that effective information technology and cybersecurity measures related to the use of distributed ledger technology should guarantee, among other things, the reliability of all smart contracts employed within the DLT market infrastructure (Recital 41 and Art. 7(4)).

[4] According to some scholars, to maximize the potential of smart contracts and at the same time guarantee the minimum level of protection of the market and their users, the European Commission has adopted a so-called "law + technology approach". See: T. Schrepel (European Commission), *Smart Contracts and the Digital Single Market Through the Lens of a "Law + Technology" Approach*, 2021.

[5] We mean here Bitcoin as a first-generation blockchain and Ethereum as a second-generation infrastructure. Third-generation blockchains have innovative features that we describe in Sections 4 and 5.

[6] We explain this model at length in Section. 5.

[7] DLT or private blockchains are characterized by a central authority that controls all operations that take place within the network. Access to the network is limited, in qualitative and/or quantitative terms, to the elements and entities authorized by the central supervisory authority. In addition, access to the transaction log and any other information is private. On a technical level, decisions regarding access and operational limits are made by a certain number of nodes, which may have specific rules for the use of the blockchain. Private DLTs/blockchains therefore guarantee protection of personal data, which does not circulate on an infrastructure accessible to all, but which has limits in terms of security, scalability, and interoperability. Public blockchains offer greater guarantees (to varying degrees, depending on the characteristics of each) in terms of security, scalability, and decentralization and are transparent by design. Logical platforms on public blockchains thus maximize the advantages of public infrastructures, while ensuring the protection of data and transactions as in private blockchains (*below*, Section 7).

[8] The contracts-on-chain model enhances the characteristics of some public blockchains, while increasing efficiency and effectiveness exponentially when embedded into a logical platform with controlled access, as we will try to demonstrate in the following paragraphs. We have conceived and developed contracts on chain since 2019 on a private blockchain (Hyperledger Fabric) to test its potential and limits. In light of the need for straightforward oversight of personal data management, which arises from centralized control, it became evident that private blockchains did not give adequate security assurances, and they fell short in terms of scalability and execution speed. Additionally, transaction costs were notably high. Consequently, in 2022, we embarked on the development of contracts on chain on a public blockchain (specifically, Algorand).

pursuit of contractual objectives while adhering to the minimal protection requirements laid out by the European legislature concerning specific types of contracts.

From this standpoint, we have chosen to apply the contracts-on-chain model to well-established contractual frameworks, with a particular focus on escrow contracts. These contractual arrangements, known for their conduct, information, and content obligations aimed to uphold the equilibrium of negotiations, are frequently standardized. We have selected these specific contractual models to facilitate a concrete comparison between current smart legal contracts and contracts-on-chain that highlights the innovative potential of the latter while keeping a line of continuity with the former.

In this article, we start from the most significant technological evolution, represented by the third-generation blockchain system, to illustrate how the features of these blockchains, especially *public blockchains*, streamline negotiation, contract conclusion, and execution processes directly on the blockchain itself, giving rise to what we name the "contract-on-chain" model/mechanism. This concept appears to offer substantial advantages, particularly from a legal perspective.

In this mechanism, the smart legal contract functions as a technical tool that allows the negotiation, conclusion, and (partial) execution of the contract, thereby reducing the risks of default and potentially resolving disputes. Moreover, it inherently links contracts and operates automatically.

From this perspective, the use of smart legal contracts allows contracts not only to be concluded on the blockchain but also to live and thrive within it, incorporating additional attributes as compared to traditionally concluded contracts. Lastly, we endeavor to elucidate the transformative potential of this latest evolution, the issues it addresses, and the opportunities it presents.

A final, methodological premise should be made explicit. This paper seeks to illustrate the potential of a tool (contracts-on-chain) within a specific technological and regulatory environment (public blockchain and a logical platform as a superstructure). We believe that this environment provides the optimal context for comprehending the effects and nurturing the innovative aspects of this tool, all while maintaining consistency with the existing tools (smart legal contracts). However, we do not intend to assert that 'contracts-on-chain' are universally suitable for all blockchain technologies or every sector.

We therefore present contracts-on-chain as a 'bridge tool' which utilizes blockchain, thus acquiring its characteristics of certainty and immutability in recording transactions, while simultaneously offering the user an experience similar to that proposed by current online contracts. Furthermore, depending on the type of contract and the parties' intentions, all or parts of the functions and activities within contracts-on-chain can pass through the blockchain. In this way, the rigidity typical of blockchain contracts is adjusted according to the specific needs of the particular case. Similarly, in contracts-on-chain, the tools used for dispute resolution can be entirely or only partially on the blockchain, again depending on the needs and intentions of the parties. For these reasons, we consider contracts-on-chain as a bridge between Web2 and Web3 that, by allowing a modular use of the blockchain, can encourage its use for contracts.

Part I – Uncertain Definitional Boundaries

1. Smart contracts (codes) and smart legal contracts

Smart contracts

According to the prevailing opinion, a smart contract is a 'computer program' that operates on a blockchain executed in a decentralized manner through the nodes of the network.[9]

In computer development environments, the notion of a 'smart contract' refers to a program that is instrumental to the automatic execution on a blockchain of a specific function that is desired and created by the programmer.[10] In this sense, smart contracts have unique characteristics as compared to any other software because (i) the program is recorded on a blockchain and acquires the characteristics of immutability, security, and transparency ;[11] (ii) the execution of the program is deterministic and the result is stored on the blockchain; (iii) the program can regulate the activities of the blockchain and therefore serve as a repository, and it can also transfer digital assets (including virtual currencies, iv) the program runs on the blockchain and remains immune to interference with its operation, provided certain characteristics of the blockchain are met. It follows that the smart contract as a 'code' does not possess the necessary characteristics to qualify it as a

[9] E. Mik, *Smart Contracts: Terminology, Technical Limitations and Real-World Complexity,* (2017) *9* Law, Innovation and Technology *269, 280*. The first definition was given by N. Szabo (*Smart Contracts: Building Blocks for Digital Markets*, 1996, 16 Extropy): a smart contract "is a set of promises, specified in digital form, including protocols within which the parties perform on these promises". M. Raskin (*Law and Legality of Smart Contracts* (2017) 1 Georgia Law Technology Review 305, p.309), defines smart contracts as "an agreement whose execution is automated" which is "effected through a computer running code that has translated legal prose not an executable program". According to K. Werbach- N. Cornell, *Contracts Ex Machina* (2017) 67 Duke Law Journal 313, smart contracts can be generally defined as self-executing digital transactions using decentralized cryptographic mechanisms for enforcement. An alternative definition is given by T. Graaf, *From old to new: from Internet to smart contracts and from people to smart contracts*, in Computer Law & Security Review, 2019 (105322). According to the A. smart contracts are software programs: 1. that are stored and executed without an intermediary in a decentralised manner on various computers (nodes) which are connected on a peer-to-peer basis to each other in a network and owned by different people; 2. that execute 'if this then that 'commands autonomously so that contractual promises are automatically executed; 3. in respect of which, as a condition precedent, a trans fer of value (e. g. payment by the customer) can only take place if ultimately at least 51 % of the nodes have reached consensus that the execution of the smart contract (e.g. provision of the service by the supplier) has occurred in accordance with the requirements stipulated in such coded contract; and 4. the storing of which takes place in a public ledger which cannot be changed and which is often referred to as a secure public ledger with a single source of truth.-. P. De Filippi- A. Wright (*Blockchain and the Law: The Rule of Code,* Harvard University Press 2018, p.74) define smart contract as an "if-then"statement that runs on the blockchain where "parties can enter into a binding commercial relationship, either entirely or partially memorialized using code, and use software to manage contractual performance."

[10] The term 'smart contract code' originally appeared in Ethereum documentation. Today, it is commonly employed in the blockchain community to describe any complex program that is stored and executed on a blockchain. While early blockchains were designed to perform a limit set of simple operations – primarily, transactions involving a currency-like token – recent technological advancements have enabled blockchains to handle more complex operations, defined in programming languages. On this subject, see also S. Nakamoto, Bitcoin: A peer-to-peer electronic cash system (2008) URL: https://bitcoin.org/bitcoin.pdf; S. Bistarelli, I. Mercanti, F. Santini, An Analysis of Non-standard Transactions, Crypto Valley Conference on Blockchain Technology (CVCBT), Zug, Switzerland, 2018, pp. 93-96, doi: 10.1109/CVCBT.2018.00016 (2018)]); G. Wood, Ethereum: A secure Decentralised Generalised Transaction Ledger, EIP-150 REVISION (2014); C. Robustella – C. E. Papadimitriu, *Reconstructive ideas on the subject of* smart *contracts, between technological innovation and legal rule* P.A. Persona e amministrazione *,* 2022, p. 963 ff.

[11] The smart contract code is saved on the ledger shared by all network participants, so it can be easily consulted and verified.

(legal) contract.[12] Those characteristics are unrelated to the primary purpose underlying the smart contract design, which is to identify a code that is stored and executed when specific conditions are met. This code might encompass a single algorithm responsible for overseeing the data flow of a company, validating account permissions, or handling questionnaire responses. In numerous instances, smart contracts do not possess an independent functionality. Instead, they serve as essential components within a larger application executed on the blockchain, thereby contributing to decentralization.

The term 'smart contract' lacks a universally accepted definition in academic discourse.[13] The efforts made to define the notion primarily revolve around a functional dichotomy between 'smart contract code' and a 'smart legal contract', as outlined in the research conducted in cooperation with the Bank of Italy.[14]

Smart legal contracts

According to an initial approach, a smart legal contract is a computer program used solely to execute, in whole or in part, a contract entered into in a traditional manner: it is a 'computerized transaction protocol' that automatically executes orders (in this case, the terms of a contract).[15] According to this theory, therefore, "a smart legal contract is not a true contract but a software (or informational protocol) developed to execute the contract".[16] However, a smart legal contract is (self-)sufficient and is programmed to execute all and only the rules incorporated into the code. It makes only the terms and conditions of the contract – as agreed upon by the parties, written in code form, and saved in the blockchain with a timestamp – verifiable, immutable, and irrevocable. When certain conditions of the agreement are met, the smart legal contract applies them (according to an "if-then" logic) and automatically produces the intended effects (i.e., approves the exchange of a token between the parties).

From a contract law perspective, the automatic execution of the smart legal contract ensures compliance with the contract's obligations: the blockchain architecture does not allow for voluntary violations of the established conditions.[17] This reconstruction entails a shift in contractual practice from an authoritative ex post judgment – typical of traditional contracts – to an automated ex ante evaluation.[18]

A second doctrinal approach is more varied within itself,[19] ranging from authors who configure the smart legal contract as a computer program used to formulate (in computer language), in whole or in part, the content of the contract, which is then executed automatically,[20] to commentators who believe it is a contract itself.[21] According to others, the smart legal contract instead designates, in general terms, a structure of the negotiating process rather than a configuration of interests.[22]

A significant and authoritative part of the doctrine questions the critical aspects of applying rules designed for traditional exchanges to

[12] See: J. Rohr, *Smart Contracts and Traditional Contract Law, or: The Law of the Vending Machine*, in 67 *Clev. St. L. Rev*. 71, 72 (2019), according to whom "Smart contract is an unfortunate name for something that is not necessarily smart, or necessarily a contract". A. Guadamuz, *All watched over by machines of loving grace: a critical look at smart contracs*, Computer Law & Security Review, 2019 10533, concludes that smart contracts are not contracts for all practical purposes. Ethereum co-founder Vitalik Buterin himself has stated that he regrets using the overly legal term "smart contract" instead of a more technical and less captivating term like "persistent scripts".

[13] The complexity of the matter extends to the extent that any attempt to provide a definition is likely to face criticism. See European Law Institute, *Principles on Blockchain Technology, Smart Contract and Consumer Protection* (2022), p. 22; M. Durovic-A. Janssen, *The Formation of Blockchain-Based Smart Contracts in the Light of Contract Law,* (2018) 26 European Review of Private Law 753.

[14] See the Bank of Italy Occasional Paper n. 863, *Characteristics of smart contracts* (M. Doria, F. Bassan, M. Rabitti, A. Sciarrone Alibrandi, U. Malvagna), July 2024.

[15] For in-depth exploration of the topic, see: , P. Cuccuru, *Beyond bitcoin: an early overview on smart contract*, in *International Journal of Law and Information Technology*, vol. XXV (2017), 179 ss; K. Kasprzyk, *The concept of smart contract from the legal perspective*, in *Review of Comparative Law* Vol. XXXIV (2018), p. 101-118; M. Raskin, *The Law and Legality of Smart Contracts*, in *Geo. L. Tech. Rev*. 305, 2017, p. 312.

[16] Rejecting a contractual nature: C. Poncibò, *The Digitalisation Of Contracts In International Trade And Finance: Comparative Law Perspectives On Smart Contracts*, in Digitalization and Firm Performance, 131 (M. Ratajczak-Mrozek – P. Marszałek eds., 2021); O. Meyer, *Stopping The Unstoppable - Termination and Unwinding of Smart Contracts*, in *Journal of European Consumer and Market Law*, 17, 19, 2020; A. Ferreira, *Regulating smart contracts: Legal revolution or simply evolution?*, in *Telecommunications Policy* 2021, p. 45.

[17] Shortcomings of smart contracts compared to traditional, court-enforced contracts are highlighted by the earliest legal doctrine: A. Savelyev, *Contract Law 2.0: 'Smart' Contracts As the Beginning of the End of Classic Contract Law,* Information & Communications Technology Law (2017) 26, 116; K. Werbach-N. Cornell, *Contracts Ex Machina* (2017) 67 Duke Law Journal 313; M.Raskin, (note 9); P. Cuccuru, *Beyond Bitcoin: an early overview on smart contracts*, International Journal of Law and Information Technology, 2017;25:179; E. Mik, (note 9); M. Giancaspro, *Is a 'Smart Contract' Really a Smart Idea? Insights from a Legal Perspective* (2017) 33 Computer Law & Security Review 825.

[18] I. Jerry, – H. Hsiao, *Smart Contract on the Blockchain – Paradigm Shift for Contract Law?*, in *US – China Contract Law Review*, 2017, p. 686 ss.

[19] Among others: P. Catchlove, *Smart Contracts: A New Era of Contract Use*, available on SSRN (3090226), p. 15; C. Clack, et al., *Smart* contract *templates: foundations, design landscape and research directions*, 2016, disponibile su: http://arxiv.org/abs/1608.00771; M. Durovic -F. Lech, *The Enforceability of Smart Contracts*, in *Italian Law Journal*, 2019, p. 504 ss; H. Eenmaa-Dimitrieva - M. Schimdt-Kessen, *Creating Markets in No-trust Environments: the Law and Economics of Smart Contracts*, in Computer Law &Security Review, pp. 69-88.

[20] Carron e Botteron, *How smart can a contract be?* in Kraus, Daniel; Obrist, Thierry; Hari, Olivier (eds), *Blockchains, Smart Contracts,* Decentralised *Autonomous Organisations and the Law* (Edward Elgar Publishing, 2019) p. 101 ss., distinguishing between a smart contract and a smart legal contract based on the moment when parties make resort to a smart contract; a smart contract is defined as being legal when it is used as a platform for an agreement and when, at the launch of the program, the parties enter into a contractual relationship with binding legal effects.The peculiarities of agreement execution, however, raise complex legal issues. A. Wright - P. De Filippi, *Decentralized Blockchain Techology* and *The Rise of Lex Cryptographia*, 2015, available on SSRN: https://ssrn.com/abstract=2580664, at pp. 24-25, admitting that "ambiguity and poor drafting can [...] be used by parties to wrestle free from contractual conditions that parties no longer want to honor […] [Parties] can use a smart contract to ensure that a contractual condition is executed, forcing the parties to remain bound to their respective obligations." Similarly: M. Durovic - F. Lech, *The Enforceability of Smart Contracts*, (note 19); M. Raskin, (note 9). Others, however, see the potential of smart contracts while recognizing the existence of challenges that need to be addressed; some are confident in the time needed for the development and continuous application of this technology until best practices are achieved. See, in this regard: E. Tjong Tjin Tai, *Force Majeure and Excuses in Smart Contracts*, in *European Review of Private Law* 2018, pp. 787-904, "smart contracts are not very well suited to deal with the finesses that are currently expected by non-lawyers and lawyers alike when it comes to excuses to performance. […] Only by extensive development of best practices is improvement to be expected". Others, instead, limit the scope of application of smart contracts to certain sectors because their limited flexibility would not allow widespread use: J. Sklaroff, *Smart Contracts and the Cost of Inflexibility*, in *University of Pennsylvania Law Review*, 2017, p. 287 ss.

[21] See: M. Maugeri, *Smart contracts and contract law,* Il Mulino, 2021.

[22] A. Benedetti, Contratto*, algoritmi e diritto civile transnazionale: cinque questioni e due scenari*, in *Riv. dir. civ.*, 2021, p. 411 ss..

smart legal contracts. Therefore, for a smart legal contract to have relevant legal effects for the legal system and be binding on the parties, its essential elements and the applicable discipline must be compatible with the civil law framework that regulates traditional contracts.[23]

Hence, 'smart contract code' and 'smart legal contracts' do not appear to fully overlap because they serve different functions.[24] Smart legal contracts, in their essence, consist of lines of code that run on the blockchain and represent an expansion of the smart contract into the legal realm. The smart legal contract is a genus that falls into the smart contract species or, to employ the set theory, it is a smaller circle wholly contained within the larger circle that represents the smart contract, specifically applied to legal relationships.

Thus, smart legal contracts "constitute a combination of programming code and legal language".[25]In legal discourse, the term smart legal contract is often understood as a tool operating on *blockchain technology* that serves to articulate, verify, and enforce agreements between parties to a contract, either as a complement to or a substitute for traditional contracts. Hence, the distinction between smart contracts (codes) and smart legal contracts has a functional character.[26] On a technical and technological level, however, the fundamental mechanism remains uniform for both categories. Blockchain technology ensures that, during the distributed execution of a smart contract, each node in the network generates identical outputs based on a given set of inputs, without having to rely on data provided by trusted third parties.[27]

With this functional perspective in mind, smart legal contracts enable the fulfillment of mutually agreed-upon contractual conditions, reduce the risk of default, and curtail the need for trusted intermediaries or traditional enforcement mechanisms.[28]

2. The life cycle of smart legal contracts

In summary, the (technological) life cycle of a smart legal contract can be segmented into four phases.

The first phase involves the translation from natural language to programming language, which entails the creation of lines of code containing the instructions for the operation of smart contracts. This process translates the contractual elements (whether essential or not) into programming language (Boolean logic). Once inscribed on the blockchain, the smart contract is immutable, unless a new version of the code is created.[29]

This phase requires a combination of technical skills, encompassing computer science (the programmer translates the content of the contract into programming language) and legal expertise (a lawyer engaged in a dialogue with the programmer so as to ensure an accurate translation of the function's content).

The second phase consists of the transcription of the code onto the blockchain. At this stage, the developer uploads the functions that she/he has configured, which the parties sign, often with an asymmetric double-key cryptographic system. The smart contract is then embedded within a block (designated by a unique *hash* code) containing other transactions and is added permanently and irrevocably to the blockchain. It is accompanied by a timestamp that explicitly indicates the date and time of the transaction. Consequently, anyone can systematically track, consistently trace, and freely access the transaction.[30] In the third

[23] The issues most discussed in doctrinal debates concern: (i) the formation and conclusion of the contract; (ii) the recognition of the parties to the agreement; (iii) the nullity of clauses; (iv) supervening events in ongoing relationships; (v) the application of general principles of the legal system; (vi) the concept of substantive justice. For further insight into this topic, see: European Law Institute, *Principles on Blockchain Technology, Smart Contracts and Consumer Protection*, 2022; P. Sirena - F. Patti, *Smart Contracts and Automation of Private Relationships*, in *Bocconi Legal Studies Research Paper Series*, 2020.

[24] The distinction is made clear by G. Jaccard, Smart Contracts and the Role of Law *(January 10, 2018),* available in SSRN: https://ssrn.com/abstract=3099885; L. Ante, *Smart Contract on the Blockchain—A Bibliometric Analysis and Review*, BRL Working Paper Series No. 10 (2020); B. Carron, V. Botteron, *How smart can a contract be?,* in D. Kraus, T. Obrist, O. Hari (eds.), Blockchains, Smart Contract, *Decentralised Autonomous Organisations and the Law*, Cheltenham, UK-Northampton, MA, USA (2019) p. 101 ff., spec. pp. 111-114; G. Rinaldi, *Smart contract: mechanization of the contract in the blockchain paradigm,* in G. Alpa (ed.), Law and artificial intelligence (2020), pp. 353-354.

[25] For a careful analysis of the combination of code and language, see the Bank of Italy Occasional Paper n. 863, *Characteristics of smart contracts* (M. Doria, F. Bassan, M. Rabitti, A. Sciarrone Alibrandi, U. Malvagna), July 2024. See also: P. Cuccuru, *Beyond bitcoin: an early overview on smart contracts*, in *International Journal of Law and Information Technology*, vol. XXV (2017), 179 ss; K. Kasprzyk, *The concept of smart contracts from the legal perspective*, in *Review of Comparative Law* Vol. XXXIV (2018), p. 101-118; M. Raskin, *The Law and Legality of Smart Contracts*, in *Geo. L. Tech. Rev.* 305, 2017, p. 312. For an experiment to translate the buyer 's suspension right into code, see T. Tjong Tjin Tai, *Formalizing contract law for smart contracts*, Tilburg Private Law Working Paper Series (2017-6).

[26] See note 16.

[27] H. Dimitrieva-M. Schmidt-Kessen, (note 19), p. 69-88, discuss how smart contracts "could provide a possible alternative mechanism for ensuring cooperation in transactions between two or more parties that cannot rely on any common legal or social background guaranteeing contract enforcement".

[28] The most recent legal doctrine recognizes that smart contracts can reduce transaction costs and increase efficiency in contracting: Cuccuru (note 15); Giancaspro (note 17). Vatiero partly disputes this assumption while also offering solutions (M. Vatiero, *Do smart contracts incur higher transaction costs than traditional contracts*? in Mathis K. and A. Tor (eds.), Law and Economics of the Digital Transformation, Springer, 2023, pp. 21-32; M. Vatiero, *Smart contracts vs incomplete contracts: A transaction cost economics viewpoint*, in Computer Law and Security Review v. 46, 105710, (2022), p. 1-8). Others highlight the potential benefities also from data and consumer protection perspective: T. Kiviat, *Beyond Bitcoin: Issues in Regulating Blockchain Transactions* (2015) 65 Duke Law Journal 569, 574; J. Fairfield, *Smart Contracts, Bitcoin Bots, and Consumer Protection* (2014) 71 Washington and Lee Law Review Online 35.

[29] Developers may decide to change approval conditions, fix code issues (bugs), or add new features. The upgrade procedure can be made arbitrarily complex. For example, to update Ethereum smart contracts, developers can perform patterns, such as contract migration and proxy patterns. However, these mechanisms vary depending on the underlying blockchain. In general, this procedure requires the implementation of governance permissions to avoid unexpected manipulation by actors who are not authorized to change the execution logic. The immutability of a smart contract is often understood as a limit with respect to the flexibility of the contract. See, among many: J. Fairfield – N. Selvadurai, *Governing the Interface Between Natural and Formal Language in Smart Contracts,* UCLA J.L. & Tech*., 2022, p. 79 ff.; M.* Giancaspro, *Is a 'Smart Contract' really a smart idea? Insights from a legal perspective*, Computer Law & Security Review, 2017, pp. 825 ff. As we will see, the contracts-on-chain solution we propose overcomes this limit.

[30] The use of blockchain also raises issues of coordination with the current European legislation on data protection (GDPR). See: M. Finck, *Smart Contracts as a Form of Solely Automated Processing Under the GDPR,* 2019, Max Planck Institute for Innovation & Competition Research Paper No. 19-01, available on SSRN: https://ssrn.com/abstract=3311370 ; C. Millard, *Blockchain and law: incompatible codes?*, in Computer Law & Security Review, 2018, p. 843-846; G. Voss, *Data Protection Issues for Smart Contracts,* in Smart Contracts: Technological, Business and Legal Perspectives (M. Corrales, M. Fenwick, S. Wrbka, eds.) 2021;

phase, when the predefined conditions are satisfied,[31] the smart contract activates and executes the designated functions by invoking transactions.

In the fourth and final phase, the smart contract is deactivated and ceases to produce on-chain or off-chain effects. Nevertheless, it remains stored within the blocks where it was initially embedded, unless its editor has included a special 'kill function' in the code and indicated the person authorized to activate this function.

As previously mentioned, the smart legal contract is exclusively composed of lines of code. For this reason, despite its ability to govern and/or automatically execute some aspects of the contractual relationships, it faces significant constraints. The most prominent of these limitations appears to be the reliance of the contracting parties on an 'expert' entrusted with the tasks of writing the code – understood as a programming language – and translating the contract drafted in natural language into this code.

To illustrate this process, we can consider an example of one particular smart legal contract among the numerous possibilities: the escrow contract. An escrow contract is an agreement between two parties, the Depositor and the Beneficiary, whereby assets such as money or deeds specified in the contract are deposited with a third party, the Depositary, as collateral. These assets are subsequently released upon the fulfillment of conditions established by the parties.

The technical transformation of the contract in the form of a smart legal contract unfolds in two phases. The negotiation and signing of the contract occur off-chain, adhering to the conventions of traditional contracts. Following this, the parties appoint expert individuals, known as developers, to place the contract on the blockchain and benefit from the advantages of self-execution. Developers are tasked with translating the contract from natural/legal language into a programming language, which enables the blockchain to automatically facilitate the execution of the contract within the parameters agreed by the parties. In essence, within the logical framework of the smart legal contract, every operation that the contracting parties can perform off-chain is converted into a specific function comprised of a series of lines of code (the so-called instructions) that are visible on the (public) blockchain as an outcome of the entire smart legal contract development process.

By way of example, we present a segment of the escrow contract and, more specifically, its function concerning the proposal for the release of funds to the Beneficiary as it appears on the blockchain.

```
   // propose_release
   proposerelease_14:
   Therefore, 1 0
   callsub stateisvalidescrow_40
   // Waiver period must be expired and there is no ongoing
release or dispute (WRONG_STATE)
   Assert
   txn Sender
   callsub ispayee_41
   // Caller must be the Payee (WRONG_SENDER)
   Assert
   intc_2 // 64
   intc_2 // 64
   ==
   // Wrong Attached hash length (expected: 64 bytes)
(WRONG_DOC_HASH_LENGTH)
   Assert
   txn Sender
   callsub setapproval_46
   bytec 11 // "attached_hash"
   frame_dig -1
   app_global_put
   bytec 17 // "release_ongoing"
   intc_1 // 1
   app_global_put
   retsub
```

3. Open issues

a) Four common issues

There are four common issues of paramount importance when it comes to the compatibility of smart legal contracts within the framework of contract law: (i) the inaccessibility of computer language to those lacking the necessary IT expertise[32]; (ii) the need to establish the degree of compatibility between the ontological rigidity of the code and the desired flexibility of the contract ;[33] (iii) the applicability to smart contracts and to smart legal contracts of safeguards provided by traditional contract law; (iv) the applicability to smart contracts and to smart legal contracts of traditional conflict-of-laws rules.

These issues are inherently objective, as exemplified by the specific case of an escrow contract. The first issue involves identifying methodologies to transform natural language, as is prevalent in a 'traditional' contract and regulatory provisions, into inputs having a binary structure. Simultaneously, it is essential to guarantee accessibility to this binary language so that the congruence of the contractual contents with the smart contract can be ascertained.[34]

(i) The inaccessibility of computer language

[31] The conditions for execution can differ: They can be on-chain, where facts can be perceived directly by the blockchain itself, such as the crediting of an amount in digital currency to a wallet connected to the smart contract. They can be off-chain, where they consist of events external to the blockchain that occur in the real world, and which can migrate and have effects on the smart contract thanks to the oracles (on oracles, see below, Section 14).

[32] See J. Allen, *Wrapped and Stacked/ 'Smart Contracts' and the Interaction of Natural and Formal Language, ERCL*, 2018, pp. 307-325; J. Fairfield – N. Selvadurai, *Governing the Interface Between Natural and Formal Language in Smart Contracts,* UCLA J.L. & Tech., 2022, at 79.

[33] J. Sklaroff, (note 20). The author observes that the lack of flexibility in smart contracts presents a major challenge to the technology's scalability. In summary, and on a critical note, the author notes that a large-scale revolution in smart contracting would introduce much higher costs than those it seeks to eliminate, regardless of whether they are left to the negotiating parties alone or shared among other stakeholders or the public in general. Scholarship has also raised a question regarding transaction costs, which are indeed relevant in first-generation (Bitcoin) and second-generation (Ethereum) blockchains. On this point, reference is made to the writings of M. Vatiero (note 28). We will see later (below, section 5, ii) how third-generation blockchains overcome this issue: it is precisely one of those cases where the technology enables the resolution of a problem (transaction costs) that would otherwise hinder blockchain development. However, the issue of transaction costs attributable to a lack of legal adaptation, as well as to poor adaptation, remains open. Nevertheless, we will see how contracts-on-chain significantly reduces them compared to the use of smart legal contracts.

[34] B. Carron - V. Botteron, *How Smart Can a Contract Be?*, pp. 115-116. The authors, regarding the issues raised by the integration of a traditional contract into a blockchain, argue that the question of interpretation and the translation of contractual plain language into computer code is difficult for at least three reasons. First, contractual language is very technical and cannot easily be replaced by commands of imperative programming (i.e., If/Then), especially when it contains indeterminate legal notions, such as termination for 'good cause', obligations of 'good faith', or a 'reasonable' period. Second, even if the contractual text could be translated into commands of imperative programming, the text does not represent the only element of legal interpretation. Third, without an automatic tool that allows for the transcription from computer code into plain language, using a smart contract for the conclusion of legal agreements would be significantly less interesting. Without such a tool, a manual and costly transcription would be required each time a difficulty arises in the performance of the contract. For the time being, programmers of translating machines may fail to appreciate the importance of every word contained in legal contracts, which may lead to disputes relating to interpretation.

The primary critical challenges concern the conversion from natural language into two machine-understandable languages: the programming language (expressed through words, numbers, punctuation marks, and other graphical symbols) and the machine language (comprised of bits conventionally represented as the numbers 0 and 1[35]); these are respectively classified as high-level and low-level languages.[36]

The programming language is devised to process instructions to be translated into machine language, which, in turn, conveys the instructions to the computers for execution.[37] An issue of adaptability arises in this context, stemming from the transposition of contractual semantics into an algorithmic key. Human language is essentially converted into programming code, which replaces the normal comprehensibility, flexibility, and versatility of natural language with binary dialectical rigidity represented by 0 and 1.[38]

The 'translation' from natural language also introduces an issue of the parties' awareness to a smart legal contract. This issue holds particular concern in certain sectors. For example, in the banking and financial sectors, transparency requirements play a pivotal role in assessing both the correctness of the intermediary's conduct and the intelligibility/comprehensibility of the agreement in terms of its content and effects.[39]

The matter of translating natural language into machine language via programming language in the context of contracts remains an unresolved issue in scholarly debates.[40]

Regardless of the flexibility guaranteed by one programming language compared to another (i.e. Phyton, JavaScript), the fundamental issue remains the significant trust that the contracting party must place in the developer who is responsible for the natural language's translation into code.[41]

To clarify, if a contract intended for execution in Chinese is drafted bilingually in English and Chinese, and one party understands English but not Chinese, the enforceability of the contract depends on the existence of a certified translation. Similarly, when a natural language contract is converted into machine language, the question arises regarding the party responsible for certifying the accuracy of the translation. This predicament, which remains unresolved thus far, casts doubt on the validity of the contributions made by legal scholars to date.

The risks associated with language and the risk of disregarding a legal system whose application could theoretically by called for by the smart contract are substantial. Many believe that this tool is unsuitable because it could end up concentrating power exclusively in the hands of those who determine and write the rules of the code. This risk is inherent in the *Code is Law* formula,[42] which has now also given rise to several corollaries, the most relevant being the 'rule of code', which either entirely or partially supplants the 'rule of law'.[43] In sum, according to the 'Code is Law' principle, the code of each blockchain is the fundamental norm dictating the legitimacy of smart contracts that can operate only subject to the correct execution of the code. This approach carries several significant consequences. Two of them are arguably most notable. First, it transforms the smart contract into the norm, the code into the fundamental norm, and the blockchain into the legal order.[44] Second, according to some legal scholars, it leads to the extreme consequence of replacing the principle of equality before the law (the rule of law) with the principle of neutrality concerning the code (the rule of code).

(ii) The relationship between the ontological rigidity of code and the desired flexibility of the contract

[35] The program written in programming language is called 'source code', while the one written in machine language and executed by the computer is the 'machine code' (or also 'object code').

[36] Smart contracts are written in a high-level programming language, e.g. Solidity for the Ethereum blockchain (see *Solidity: Ethereum Smart Contracts Programming Language*, https://soliditylang.org/), or a scripting language, e.g. Bitcoin Scripting (see: S. Nakamoto, Bitcoin*: A peer-to-peer electronic cash system*, and S. Bistarelli - I. Mercanti - F. Santini, *An Analysis of Non-standard Transactions*, Frontiers in Blockchain, 2019, pp. 93-96), and are compiled into a set of bytecode instructions. The compiled bytecode of a smart contract is installed within an execution environment. There are several execution environments that generally fall into two categories: (i) memory stack-based interpreters (e.g., Bitcoin Script interpreter) (ii) virtual machine-based interpreters (e.g., Ethereum Virtual Machine). See: V. Buterin, *A Next Generation Smart Contract & Decentralized Application Platform*; N. Kannengießer, S. Lins, C. Sander, K. Winter, H. Frey, A. Sunyaev, *Challenges and Common Solutions in Smart Contract Development,* IEEE Transactions on Software Engineering*,* 2022, pp. 4291-4318; G. Wood, *Ethereum: A Secure Decentralised Generalised Transaction Ledger* Berlin Version*;* W. Zou, *Smart Contract Development: Challenges and Opportunities,* IEEE Transactions on Software Engineering*, 2021, pp. 2084-2106.*

[37] To convey the instructions initially conceptualized in natural language to the computer, a two-step translation process is required. First, the instructions must be translated into the programming language (initial translation). Then, in a second phase, this programming language is automatically translated into machine language, which is performed by computers, known as compilers, that use specialized programs created for this purpose.

[38] J. Sklaroff, (note 20).

[39] See: Bank of Italy Occasional Paper n. 863, *Characteristics of smart contracts* (M. Doria, F. Bassan, M. Rabitti, A. Sciarrone Alibrandi, U. Malvagna), July 2024.

[40] One the one side, computer language eliminates the possibility of a flexible interpretation of contract performance, (i.e. duties of good faith or standards of best efforts). On the other side, encoding of smart contracts carries the risk that the code might not run as envisaged or contain bugs that will result in malfunctions. See: Cuccuru (note 15), Giancaspro (note 17), Werbach-Cornell (note 9), Raskin (note 14); Mik (note 9).

[41] A solution to this problem, which is also interesting because it uses technology as a regulatory tool, thus applying "participatory regulation" (note 58), is the certification of translation, described in detail by: J. Krijnen-M. Chakravarty-G. Keller-W. Swierstra, *Translation certification for smart contracts*, in Computer Law & Security Review, 2023.

[42] L. Lessig, *Code version 2.0*, Basic Books, 2006; L. Lessig, *Law Regulating Code Regulating Law*, in *Loyola University Chicago Law Journal* 2003, pp. 8 ss.; L. Lessig, *Code is law*, Harvard Magazine, 1.1.2000.

[43] P. De Filippi, M. Mannan, W. Reijers*, Blockchain Technology and the Rule of Code: Regulation via Governance*, 2023, available at: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4292265. The authors compare the internet and blockchain technology as technologies that tend to resist traditional regulation, thereby introducing the concept of 'rule of code' as a new regulatory principle introduced by blockchain technology. This notion distinguishes itself both from the rule by code adopted by large internet platforms and the rule of law endorsed by states. The rule of code is used to stress that technological arrangements can be designed in such a way as to eliminate—or at least reduce—the arbitrary influence of any single actor (including the state) over the operations of a technological system, as no individual actor can unilaterally dictate actions or changes to the blockchain network, including core developers. In other words, through a constitutional lens, no actor has a claim to sovereign authority over the network.

[44] In this reconstruction, the smart contract is the norm that operates using a code (and whose rules must therefore be respected) within a technological system (the blockchain). When smart legal contracts, which define the legal relationships between two or more parties, operate on the blockchain, the latter also acquires legal significance. Considering that the blockchain operates as an autonomous infrastructure with its unique set of rules to which smart contracts must adhere, it can be legally classified as an autonomous legal system. Each blockchain system interacts with others, which introduces the critical concern of interoperability. Smart contracts specifically designed for one blockchain may not be applicable on different blockchain platforms. While current technological advancements have addressed this issue with progressively sophisticated solutions, they are not yet completely satisfactory.

On a broader scale, the issue arising from the absence of natural language in smart legal contracts, often referred to as the 'dark side' of these contracts, appears insurmountable. Moreover, from a different perspective – and to frame it in a language more in line with the sensibility of legal positivists – a thorny question arises concerning the extent to which the automation of smart legal contracts neutralizes interpretational flexibility in negotiation rules and adaptability to changing circumstances.

It is questioned in particular to what extent the automatism of a smart legal contract excludes flexibility in the interpretation of the contractual rule, the adaptability of the agreement to changing circumstances ('efficient breach'), and remedies, should the performance go wrong (as in cases of hard forks).

This aspect has garnered great attention in the international debate.[45] It is worth noting in this regard that as the provisions within a smart legal contract are not interpreted in accord with the law or the will of the parties and are instead governed by the code of the smart contract, the latter might execute a specific set of conditions defined by the code, even if the contract initially intended by the parties necessitated a different type of performance based on their will and the circumstances.

As a result, the execution of smart legal contracts could create a mismatch between the will of the parties, the provisions established by the traditional legal order (in accord with contract law), and the conditions established by the technological infrastructure of a blockchain (in accord with its underlying protocol and the smart contract). In this context, legal scholarship distinguishes between regulation by law and regulation by code as regards their inherent characteristics, such as natural language vs. formal computable language, amendability vs. immutability, and *ex-post* application by third parties vs. the absence of *ex-post* oversight.

These aspects bear great theoretical and practical importance, but at the same time they prove challenging to resolve. However, in the subsequent sections of this investigation, we will endeavor to chart a path that aligns with the current state of the art and to establish whether there is a way to find a meeting point between regulation by law and regulation by code, thus overcoming the opposition that exists between the two approaches.[46]

(iii) The applicability to smart contracts and to smart legal contracts of safeguards provided by traditional contract law

From the anonymity of the parties, which blockchain enables, legal doctrine derives certain consequences related to the inability to apply to smart contracts and to smart legal contracts the safeguards provided by traditional contract law to lack of capacity, or to violations of mandatory rules regarding duress, mistake, or fraud. There are also no safeguards against illegal smart contracts (whose object or effect is against the law).[47]

This assumption, however, does not hold true for private blockchains, where access is restricted and verifying the identity of the parties is (or should be) an essential requirement. For public blockchains, the issue of potential anonymity does arise. To address this challenge, we propose the use of logical platforms (see section V).

(iv) The applicability to smart contracts and to smart legal contracts of traditional conflict-of-laws rules

In the realm of private international law, the issue raised by smart contracts and smart legal contracts is whether traditional conflict-of-law rules are adequate. If they are not, it becomes necessary to determine which new rules should be introduced.

The prevailing legal doctrine holds that the current connecting factors (party autonomy, the law of the closest connection, the *lex loci contractus*, and the *lex rei sitae*) can be applied to smart legal contracts and are sufficient.[48] For the smart contract code, the *lex loci solutionis* would apply, which refers to the place of performance of a smart contract used as a tool in a DLT context.[49]

However, in practice, it is difficult to apply the law of the closest connection to smart legal contracts when the parties are not identified and their residence is unknown, or when the subject of the transaction is natively digital and thus not tied to a specific location.[50] Furthermore, for the smart contract code, identifying the place of performance within the blockchain (especially if it is public) seems to be challenging.

Overriding mandatory rules[51] can be applied to smart legal contracts, much like they are applied to functionally equivalent legal contracts outside the DLT. Examples include rules for contracts involving weaker parties, such as consumer or insurance contracts, or rules prohibiting money laundering, terrorism financing, or tax evasion, as well as regulatory rules aimed at maintaining the stability of the financial system. However, applying these rules to smart contract code is more complex. Nevertheless, the a-territorial nature of decentralized blockchain (especially if public) raises significant enforcement issues.

Some of the issues of private international law are addressed by the control that blockchain enables: directly, in the case of private blockchains, and indirectly—via logical platforms—in the case of public blockchains. This structure allows us, on public blockchains, to replace the decentralization typically associated with blockchain with a distributed model. In this model, various logical platforms—which users access upon identification and, if necessary, KYC (Know Your Customer), AML (Anti-Money Laundering), etc.—are interconnected.

[45] For an up-to-date overview of the debate, see: M. Blaszczyk, *Smart Contracts, Lex Cryptographia, and Transnational Contract Theory*. Available at SSRN: https://ssrn.com/abstract=4319654 or https://doi.org/10.2139/ssrn.4319654.

[46] P. De Filippi, M. Mannan, W. Reijers, (note 43), begin by comparing the internet and blockchain technology as technologies that resist traditional regulation in order to introduce the notion of the rule of code as an alternative to the notion of the rule of law: "We refer here to the rule of code as a new regulatory principle introduced by blockchain technology, which distinguishes itself both from the rule by code enacted by large Internet platforms, and the rule of law endorsed by states. […] The rule of code is used to stress the fact that technological arrangements can be designed in such a way as to eliminate—or, at least, reduce—the arbitrary influence of any single actor (including the state) over the operations of a technological system as no individual actor can unilaterally dictate actions or changes to the blockchain network, including core developers. In other words, if we continue to use a constitutional lens, no actor has a claim to sovereign authority over the network."

[47] See: Cuccuru (note 15); Giancaspro (note 17); Werbach-Cornell (note 9);

[48] M. El Harrak, *Do smart contracts need new conflict-of-law rules?*, in (Ed. by A. Bonomi – M. Lehmann – S. Lalani) Blockchain and Private International Law, 2023, p. 479-493; A. Held, *Crypto Assets and decentralised ledgers: does situs actually matter?* in (Ed. by A. Bonomi – M. Lehmann – S. Lalani) Blockchain and Private International Law, 2023, p. 479-493. 209-257; M. Haentjens – M. Lehmann, *The law governing secured transactions in digital assets*, in (Ed. by A. Bonomi – M. Lehmann – S. Lalani) Blockchain and Private International Law, 2023, p. 456-477.

[49] Pursuant to Article 12(2) of the Rome I Regulation, which governs the choice of law in the European Union (Regulation (EC) 593/2008 of the European Parliament and the Council of 17 June 2008 on the law applicable to contractual obligations-Rome I) "in relation to the manner of performance and the steps to be taken in the event of defective performance, regard shall be had to the law of the country in which performance takes place".

[50] A Kleczewski, *The good, the bad and the ugly: the Private International Law, the crypto transactions and the pseudonyms*, in (Ed. by A. Bonomi – M. Lehmann – S. Lalani) Blockchain and Private International Law, 2023, p. 128-155.

[51] Pursuant to Article 9.1 of the Rome I Regulation, overriding mandatory rules are "provisions the respect for which is regarded as crucial by a country for safeguarding its public interests, such as its political, social or economic organisation, to such an extent that they are applicable to any situation falling within their scope, irrespective of the law otherwise applicable to the contract".

b) How EU Regulation is addressing the open issues

Some relevant aspects of smart contracts are beginning to be addressed by EU Regulations on market infrastructure (DLT Pilot[52]), and on cryptoassets (MICA[53]). Furthermore, the Data Act[54] defines smart contracts[55] and identifies specific requirements that they must comply with regarding interoperability[56] and data sharing.[57]

However, the European regulations do not address the three aspects of smart contracts highlighted here, which are probably the most critical. Thus, standards, guidelines, and codes of conduct are beginning to fill the implementation gap left by the primary level regulations. This necessarily has to occur at a supranational level, also due to the private international law issues that smart legal contracts raise.

The approach by which regulation proceeds is that of "participatory regulation", where regulators and the market work together from the initial stages of product and service development.[58] The European Union represents the outpost of this evolution: primary level regulations (such as the Data Act, the DLT Pilot, MiCA) are implemented through standards and guidelines adopted by European Standardization Organizations (CEN, CENELEC, ETSI), by European regulators (EBA, ESMA, ECB), and by national regulators (National Central Banks, National Financial Markets Authorities), which are developed together with operators, also using tools such as sandboxes[59] and pilots. In some cases, international collaboration is more extensive: the Principles on Digital Assets and Private Law adopted by UNIDROIT in 2023 represent a significant source at present, and hopefully will be implemented by member states.[60]

Thus, the development of "participatory regulation by technology"[61] is becoming increasingly significant. Technology is not just the problem; it can also be part of the solution: it enables the provision of guarantees that regulators otherwise could not offer. In this way, technology becomes a tool for regulation.

c) How contracts-on-chain addresses the open issues

Contracts-on-chain aligns with this line of research and development: the technology used allows, on one hand, to address some of the questions raised by legal doctrine—regarding the translatability of languages, the rigidity of the tool, and its a-territorial nature—and, on the other hand, to provide solutions that comply with the demands of regulators. This includes, for example, ensuring technological development that is consistent with both the current guarantees for contracting parties by contract law and the protections for consumers or otherwise weaker contracting parties (overriding mandatory rules) .[62]

Due to its flexibility, as it allows the use of blockchain for all contractual phases (KYC, negotiation, stipulation, execution, dispute resolution) or only for some of them, based on the rules imposed by legislators, the guidelines set by regulators, and also the will of the parties, contracts-on-chain—especially when integrated into a logical platform operating on a blockchain—is a useful tool to accelerate the use of blockchain in the field of contracts.

Part II – From Smart Legal Contracts to Contracts-on-chain

4. The underlying need

The underlying idea of this study is that several of the issues raised by legal scholars can find solutions through an exploration of the latest

[52] The Regulation (EU) 2022/858 of the European Parliament and of the Council of 30 May 2022 on a pilot regime for market infrastructures based on distributed ledger technology, and amending Regulations (EU) No 600/2014 and (EU) No 909/2014 and Directive 2014/65/EU (DLT Pilot Regime), provides the legal framework for financial services based on DLT technology. The aim is to remove regulatory barriers to the issuance, trading and settlement of financial instruments issued in digital form and to support regulators in gaining experience in the use of DLT.

[53] Regulation (EU) 2023/1114 of the European Parliament and of the Council of 31 May 2023 on markets in crypto-assets, and amending Regulations (EU) No 1093/2010 and (EU) No 1095/2010 and Directives 2013/36/EU and (EU) 2019/1937 (MiCA Regulation).

[54] Regulation (EU) 2023/2854 of the European Parliament and of the Council of 13 December 2023 on harmonised rules on fair access to and use of data and amending Regulation (EU) 2017/2394 and Directive (EU) 2020/1828 (Data Act).

[55] A smart contract is a "computer program stored in an electronic register system in which the outcome of the execution of the program is recorded in the electronic register" (art. 2.39), considered a tool potentially capable of "providing data holders and recipients with guarantees of compliance with the conditions for data sharing" (par. 1 of the Report - "Background of the proposal - Reasons and objectives of the proposal").

[56] As for interoperability, Article 33 of the Data Act requires participants in data spaces that offer data or data services to other participants to provide "the means to enable the interoperability of tools for automating the execution of data sharing agreements, such as smart contracts shall be provided" (Article 33 (1)(d)). The provision provides for a presumption of conformity for smart contracts that meet the conditions set out in harmonised standards adopted by European standardisation organisations at the request of the Commission, in accordance with the European Standardisation Regulation (Regulation (EU) No 1025/2012). Moreover, in the absence of such harmonised rules, it is provided that the Commission may adopt, by means of implementing acts, common specifications relating to each requirement referred to in paragraph 1.

[57] As for data sharing, Article 36 of the Data Act is addressed to "the vendor of applications using smart contracts or, in absence thereof, [to] the person whose trade, business or profession involves the deployment of smart contracts for others in the context of executing an agreement or part of it to make data available". These entities must ensure that the smart contract complies with four key characteristics:(a) robustness: it must have been designed in such a way as to offer access control mechanisms and a very high degree of robustness in order to avoid functional errors and to resist manipulation by third parties (Article 36(1)(a));(b) safe termination and interruption: it must provide for a mechanism to interrupt the continuous execution of transactions. In particular, "[the smart contract must] include internal functions which can reset or instruct the contract to stop or interrupt the operation, in particular to avoid future accidental executions" (Article 36(1)(b));c) archiving and continuity of data: in the event that it is necessary to proceed with the termination or deactivation of a smart contract, it is necessary to provide the "possibility to archive the transactional data, smart contract logic and code in order to keep the record of operations performed on the data in the past (auditability)" (art. 36, par. 1, letter c);d) access control: a smart contract must be protected by means of strict access control mechanisms at the level of governance and the smart contract itself (Article 36(1)(d));e) consistency: a smart contract must be consistent with the terms of the data sharing agreement that the smart contract executes (Article 36(1)(e)).Seller and/or entrepreneur or professional are responsible for ensuring that the EU declaration of conformity complies with the essential requirements of art. 36, par. 1 of the Data Act. The standard also introduces a presumption of conformity for smart contracts that meet the harmonised standards adopted in accordance with the rules of the European Standardisation Regulation which impose requirements similar to those of art. 36, par. 1 (art. 36, par. 4).

[58] F. Bassan, *Digital Platforms and Blockchains: The Age of Participatory Regulation*, in European Business Law Review 34, no. 7 (2023): 1103-1132.

[59] On 14 February 2023, the European Commission launched a European Blockchain Regulatory Sandbox for innovative use cases involving Distributed Ledger Technologies and/or Blockchains. The initiative is based on the need to overcome the current legal uncertainty, caused by a complex governance of the process. The new methodology aims to simplify and strengthen the dialogue between regulators and innovators. Specifically, the Commission identifies regulatory barriers to the roll-out of solutions, and provides advice, expertise and regulatory guidance in a safe and secure environment for participants. In essence, the Commission will make use of the operators to deepen the technical aspects of these technologies, while the operators will contribute to identifying the best practices for the market, according to the process of participatory regulation.

[60] UNIDROIT Digital Assets and Private Law Principle, 2023.

[61] F. Bassan, (note 58).

[62] E. Mik (note 9).

technological advancements. It is precisely the ongoing technological evolution that provides the answers to many of the inquiries posed.

Indeed, third-generation public blockchains possess characteristics that distinguish them from earlier-generation blockchains in terms of quality.[63] These advancements are designed to enhance the efficiency of the infrastructure, which in turn allows the use of a range of tools that, for the first time, bring blockchain closer to real-world applications. As for contracts, third-generation public blockchains now blend traditional contracts with smart legal contracts, the latter being instrumental to the contract's execution. This approach ensures a user experience resembling that of online contracts in natural language, while simultaneously offering the robust guarantees inherent to blockchain technology.

In other words our starting point is the need to avoid considering smart legal contracts in isolation. In both legal theory and practical IT implementation, the concept of crafting digital contracts that encompass smart contracts is already widely accepted.[64] However, a more significant integration between the two can be realized by reversing the perspective.

Resorting to blockchain in conjunction with smart legal contracts as a procedure lays the groundwork for the following actions: i) concluding a digital contract in natural language by negotiating the content directly on the blockchain; ii) subsequently uploading it as a hashed document onto the blockchain, accessible solely by cryptographic key holders; (iii) executing it in part through the underlying technology of smart legal contracts; iv) organically interlinking it with other contracts.

In summary, the outcome is a contract concluded on a public blockchain instead of the (digital) internet, whereby the contract inherits the characteristics of the blockchain where it is conceived, developed, and used. This approach allows users to directly design the text of the contract.

This new model empowers the parties to negotiate, sign, and execute – in whole or in part, depending on both the type of contract and the parties' needs – a contract written in natural language directly on the blockchain, not just on the internet as in the case of a normal online contract.

Through this approach, the entirety of the interactions between the parties is recorded and executed on the blockchain if so desired.

In comparison to the online contracts from which they inherit the user experience, contracts-on-chain incorporate 'structural' elements that derive from the blockchain on which they operate. This grants them attributes of certainty and security which are contingent on the blockchain itself, while retaining the knowledge and awareness of the parties that is derived from the use of natural language.

Furthermore, the connecting component between natural language and the blockchain, which is the executed software, remains transparent. This software is documented on a public blockchain and can be accessed through a standard browsing tool on the chain (such as an explorer) by the relevant parties, interpreters, regulators, arbitrators, or judges, depending on the circumstances and entities involved.

As elaborated further below, the software executed on the public blockchain is accessible and easy to understand, even to a legal professional or anyone tasked with overseeing the activities of private individuals. Thus, these individuals can directly verify whether the will of the parties has been accurately registered and executed, or they can opt to engage a third party (certifier) for this purpose.

In summary, with the formula 'contracts-on-chain', we refer both to the negotiation process and the outcome of such negotiation. These contracts, akin to online contracts (and benefitting from the established user experience), use blockchain technology via a software design that integrates smart contracts so as to acquire and/or strengthen contractual certainty in the negotiation, underwriting, and enforcement stages. The blockchain – on which the contract is composed, discussed, formalized, and executed – further emphasizes the value of the negotiation.

The present investigation will now delve deeper into the 'contracts-on-chain' realm.

5. The logic platform on 3rd generation public blockchains

One of the ramifications resulting from the attributes of the latest generation of public blockchains is the inherent transparency of recorded transactions: once they are entered into the ledger, they become visible to all parties. Furthermore, in instances where decentralization is genuinely achieved, it precludes any external oversight over activities or transactions.

These characteristics have prompted certain scholars, particularly those engaged in the world of DAOs (Decentralized Autonomous Organizations),[65] to speculate on a novel mode of governance, one which either replaces or complements traditional governance and which has the potential to democratize not only finance but also the entire society owing to the decentralized structure on which it is exercised.

[63] As specified in note 9, we started developing contracts-on-chain in 2019 on a private blockchain (*Hyperledger Fabric)*. Later, in 2022, we opted for a public blockchain *(*Algorand), mainly because third-generation public blockchains have characteristics that overcome most of the limitations of private blockchains and, furthermore, they reduce or eliminate, with appropriate precautions, the drawbacks of second-generation public blockchains.

[64] See the 'ELI Principles on Blockchain Technology, Smart Contracts and Consumer Protection' (European Law Institute, 2022. Some models can be traced back to the Ricardian contract, a model consisting of a single document, written in both semantic and computer language such that it can be contextually understood by both man and machine (machine readable). This so-called hybrid model is hypothesized by I. Grigg, *The Ricardian Contract,* Proceedings of the First IEEE Workshop on Electronic Contracting*, 2004, as well as* Id. *Why the Ricardian Contract Came About? A Retrospective Dialogue with Lawyers*, in J.G. Allen - P. Hunn (eds.), *Smart Legal Contracts: Computable Law in Theory and Practice*, Oxford, 2022, p. 88 et seq.

[65] A DAO should not be understood as a blockchain network in and of itself but rather as an organization which deploys smart contracts on top of an existing blockchain network. They have relatively recent origins. For an in-depth analysis, see S. Hassan, P. De Filippi, *Decentralized Autonomous Organizations,* Internet Policy Review*,* 2021, pp. 10 ff., who describe a DAO as a blockchain-based system that enables people to coordinate and govern themselves as mediated by a set of self-executing rules deployed on a public blockchain, and whose governance is decentralized (i.e., independent from central control). See also B. Carron, V. Botteron, *How Smart Can a Contract Be? supra.* The authors state that each member of a DAO contributes by bringing his own efforts or resources, such as cryptocurrencies, in exchange for tokens. The member then participates in the decision-making process within the DAO. A DAO functions without the need for a management team and can be directly governed by its members according to the rules encrypted in the code. Similarly: N. Augustin - A. Eckhardt - A. Willem de Jong, *Understanding decentralized autonomous organizations from the inside,* Spriger, 2023. Today, DAOs relate to a wide array of structures. In terms of governance, some scholars have focused on the limitations and challenges that arise from using this structure over blockchain technology: J. Z Garrod*, The real world of the decentralized autonomous society*, TripleC: Communication, Capitalism & Critique, 2016, pp. 62–77; U. Chohan, *The Decentralized Autonomous Organization and Governance Issues* (Notes on the 21st Century) [Discussion Paper], University of New South Wales*,* 2017, available on https://doi.org/10.2139/ssrn.3082 055; K. T. Minn, *Towards Enhanced Oversight of "Self-Governing" Decentralized Autonomous Organizations: Case Study of the DAO and Its Shortcomings,* NYU J. Intell. Prop. & Ent. L, 2019, pp. 139 ff.; M. Hütten, *The soft spot of hard code: Blockchain technology, network governance and pitfalls of technological utopianism,* Global Networks*, 2019*, pp. 329–348. Others have highlighted the opportunities that the use of DAOs offers: D. Rozas - A. Tenorio-Fornés - S. Díaz-Molina - S. Hassan, *When Ostrom Meets Blockchain: Exploring the Potentials of Blockchain for Commons Governance,* 2018, available on https://eprints.ucm.es/id/eprint/5964 3/1/SSRN-id3272329.pdf; Y. Hsieh - J. Vergne - P. Anderson - K. Lakhani - M. Reitzig, *Bitcoin and the rise of decentralized autonomous organizations,* Journal of Organization Design*, 2018, pp.1–16; K.* Jones, *Blockchain in or as governance? Evolutions in experimentation, social impacts, and prefigurative practice in the blockchain and DAO space*, in *Information Polity*, 2019, pp. 469–486.

Every time a piece of data is recorded on the blockchain, it is not stored in a singular location but is rather simultaneously distributed across all nodes within – depending on the chain – a more or less secure, universal, encrypted, and, indeed, more or less decentralized ledger. This approach serves to eliminate intermediary costs, bolster security, and prevent the consolidation of power in the hands of a few.

At present, particularly in the financial sector, true decentralization still appears to be a distant goal. While digital innovation is advancing rapidly, the digital transition to a fully decentralized environment is not occurring at the same pace. Confidence in new technologies among companies and markets is increasing at a slower rate, and it takes time for regulators, markets, and users to develop trust in new governing processes.

Notably, although the absence of stringent regulations in the European Union serves as a valuable incentive promoting innovation, the lack of a minimal and shared level of regulation in the decentralized finance (De-Fi) sector indeed hinders market consolidation. In various sectors, such as banking, finance, and insurance, where blockchain can develop the most relevant applications, there are instances where an intermediary or supervisory role is still necessary, as mandated by regulations.

This need to oversee economic initiatives and the technology itself, along with its implications, has led the market to establish 'logical platforms with controlled access' on public blockchains, or to resort to private blockchains, which, however, do not offer the advantages – mentioned above - typical of public blockchains.[66]

i) Logic platforms

By 'logic platforms with controlled access', we mean digital infrastructures capable of connecting different systems and making them available on one or more public blockchains, through simplified interfaces. Access is granted to authorized users following an identification and validation procedure, and users can thus access the products and services offered by the platform.

Logical platforms with controlled access leverage *public* blockchains, while retaining all the inherent advantages of blockchains. Operators can more easily comply with the rules of regulated markets and avail themselves of control mechanisms that enhance the level of security and that make public blockchains them similar to private blockchains in terms of protection. Access to these logical platforms on the public blockchain is subject to, among other things, completing a KYC (know your customer) procedure, filling out due diligence and AML (anti-money laundering) forms, and using non-anonymous wallets provided by the platform.[67] To this end, the platform operators implement whitelisting strategies to ensure that access is granted exclusively to pre-approved crypto addresses. Subsequently, they proceed to verify compliance with the various legal rules imposed by industry regulations, as in the case of anti-money laundering.

Moreover, this process serves the dual purpose of ensuring that intermediaries, customers, and incoming funds are pre-verified, thus guaranteeing that all expressions of intent and transactions – including financial ones that are recorded and executed on the public blockchain – remain securely, permanently, and immutably traced. As we will endeavor to demonstrate in this study, controlled-access logical platforms operating on public blockchains present an appropriate solution for the ongoing transition from Web2 to Web3 and will prove valuable in subsequent phases, when fully operational, in situations where DAO models (completely decentralized platforms) are unsuitable or incompatible with regulatory prerequisites.

In this setting, the logical platform that offers contracts-on-chain allows for the inclusion of a contract text that parties can negotiate on the blockchain, following a process similar to the one underlying online contracts. Alternatively, they can conclude and execute the contract on the blockchain, blending some aspects of traditional contract flexibility with the advantages offered by public blockchain, including enhanced certainty and security. This facet pertains to the freedom of the parties to determine the content of the contract and select the method of its conclusion, provided the contract is lawful, valid, and worthy of protection under the relevant legal system.

On closer inspection, contracts-on-chain can also be used in a *private* blockchain environment (as we have experimented since 2019 with Hyperledger Fabric – *supra*, note 9). Likewise, and in general, these considerations apply to public blockchain environments that lack a controlled access logical platform. For the sake of analytical simplicity, throughout this discussion we assume that contracts-on-chain are developed within a controlled access logical platform.

This choice aligns with our intention to support the ongoing technological transition and provides more detailed analyses and solutions within an environment and a model poised to gain prominence in the market.

(ii) 3rd generation public blockchains

The technological properties of third-generation public blockchains assume a crucial relevance for the development of contracts-on-chain, as the technology provides some of the necessary guarantees to enable the integration/hybridization between online contracts written in natural

[66] On the differences between *private* and *public* blockchains, see *supra,* Section. 4.

[67] These procedures allow the platform operators to provide for all forms of oversight typical of a *private* blockchain, while overcoming its typical limits. The result is a form of regulation over "entry to the platform" that is consistent with current industry regulations, combined with a level of security, speed, scalability, and cost-effectiveness guaranteed by the *public blockchain.*

language and smart legal contracts.[68] The most important characteristics concern energy efficiency, transaction costs, scalability, decentralization, security, transaction finality, the impossibility of forks,[69] and the contextuality of the exchange between performance and consideration.

In particular: the speed of execution and recording on the blockchain allows for real-time negotiation between parties.[70] Notarization costs are minimal, also due to the exponential reduction in energy consumption, allowing for the recording of every step, including negotiation.[71] High scalability (measured by the number of possible transactions per second) allows for massive utilization of the tool, even for payment purposes.[72] Delivery vs payment (DvP) mechanisms ensure the contextuality between performance and consideration on the chain, significantly reducing (and in some cases eliminating) the risk of contractual default or counterparty risk.[73] The immediate finality of the transaction, established once recorded on the chain, safeguards parties against potential compromise of any block in the chain, preventing – ontologically – the possibility of forks or chain splits, thus eliminating the risk of a transaction or a manifestation of will being simultaneously on two different branches of the chain, directed at two different parties. The registry is unique and cannot be modified retroactively.[74] The level of security of third-generation public blockchains,[75] determined by both the degree of decentralization and the sophisticated encryption utilized (which is also resistant, prospectively, to the challenge of quantum computers), ensures integrity and certainty for these blockchains.[76]

Part III - Contracts-on-chain: smart legal contracts and public chain. An empirical investigation

The theoretical system explored so far has been put to the test through an empirical investigation involving a third-generation blockchain with the aim of addressing various questions posed by legal literature that have yet to find practical solutions. These questions encompass topics such as the translation of natural language into machine language, the relevance of legal design, and the use of blockchain for the execution phase of contracts as well as for establishing consent or even formation of the contract. Additionally, the investigation explores the possibility of anchoring the contract within a legal system, without

[68] H. Dimitrieva-M. Schmidt-Kessen, (note 19, at p. 70) correctly stated that: "the design of the underlying technology (at least in terms of the identifiability of persons transacting on blockchain, the selection of nodes and the size of network, the particularities of the consensus mechanism and the transparency of the content of the blocks) is a significant feature that needs to be factored into the discussion. This is a point often overlooked in legal academic literature".

[69] As for the Algorand blockchain that we used, see: J. Chen, S. Micali, *Algorand: A secure and efficient distributed ledger*, in THEORETICAL COMPUTER SCIENCE, v. 777 (2019) p. 155-183, and Y. Gilad, R. Hemo, S. Micali, G. Vlachos, N. Zeldovich, *Algorand: Scaling Byzantine Agreements for Cryptocurrencies*, MIT CSAIL (https://ia.cr/2017/454). Algorand has build a chain that never forks. More specifically, the forking probability is 10−18; in essence, if there was one block per second since the big bang, the chain would have probably forked once. This is a very clever way to stop the Nothing at Stake problem - if the chain can't fork, there aren't multiple competing chains for validators to validate, and the network remains stable.Algorand builds this "unforkable chain" through a handful of cryptographic techniques (described in the two articles quoted above). The basic idea is as follows:• One randomly selects one user to propose the next block, with the likelihood of selection based on ownership of the cryptocurrency, Algos.• Then, one randomly selects 1000 users to vote on validating the next block, again with the likelihood of selection based on Algo ownership.• A majority of those users need to validate that block using an "ephemeral" secret cryptographic key, which they immediately delete after validation. This prevents two blocks from both being verified simultaneously. One of the things that made Algorand unique when it was first described was its ability to select 1000 users in a decentralized way. They use a technique called 'cryptographic sortition', which is a way of describing a fully decentralized lottery. This is done through a cryptographic primitive called a 'verifiable random function'. A VRF is essentially a (pseudo-)random number generator, but unlike conventional PRNGs, it also generates a proof that the random output was generated correctly. Thus, to be selected to participate in the voting committee, a user simply needs to get a lucky draw from their VRF. The fact that it's verifiable means that users can't lie and pretend that they got lucky; this gives us a secure, verifiable set of 1000 validators.How does this result in a 10−18 forking probability? Algorand has one major assumption built into it. For this to work, 2⁄3 of all Algos, the chain's currency, needs to be controlled by "honest money", parties who always obey the rules of the Algorand protocol without fail. This is vast oversimplification, but the basic idea is that it would be very unlikely for a user with <1⁄3 control of the money supply to not only be selected to propose a block, but also be a majority of the verification committee. Plus, there are some additional Byzantine Agreement protocols that make it even harder for an attacker to wrest control of the network, which bring the fork probability down to that impressively low value.

[70] Today, a new block is inserted into the chain every 3.2 seconds (ref: Algorand).

[71] The costs of each transaction on the Algorand blockchain are equal to €0.00001. As for energy efficiency, the shift from proof of work (Bitcoin, Ethereum before the merge) to proof of stake, which characterizes third-generation blockchains, allows for a reduction in energy consumption by several orders of magnitude. The most technologically advanced blockchains have an energy consumption equivalent to that of three houses (in contrast, the Bitcoin blockchain still consumes as much as the entire Netherlands). For further insight into transaction costs, see: A. Delgado De Molina Rius, *Smart Contracts: Taxonomy, Transaction Costs, and Design Trade-offs*, in J .G. Allen e P. Hunn (Editors), *Smart Legal Contracts: Computable Law in Theory and Practice*, Oxford, 2022, p. 107 e s. e, spec., p. 121 e s.

[72] The scalability of the latest generation blockchains is significant: in fact, it goes from 16 operations per second (Ethereum) to 10,000 (Algorand) but potentially up to 40,000; to be clear, the latter relates to the order of magnitude of the exchanges of current credit card systems.

[73] As for the trade-off between performance and counterpart performance, and thus the conclusion of the agreement, the third generation of blockchains (we have experimented on Algorand) ensures contextuality. Performance (e.g., selling an asset) and counterpart performance (e.g., payment of the price) occur in a single block of the chain (atomic swap). If the two performances are not recorded in the same block, the operation (e.g., the sale of an asset) does not occur. The operational tools of the blockchain thus serve as a guarantee against the risks of non-performance (of asset delivery and payment of the price).

[74] First and second-generation blockchains required, for the completion of a transaction, the generation of several blocks on the chain following the one in which the transaction was inserted. This was to ensure a 'reasonable probability' that the operation would be inserted into a block located in the 'correct branch' of the chain, in the event of a fork. Some next-generation blockchains instead ensure the immediate finality of the transaction – definitively once it is recorded on the chain – and are therefore compliant with European regulations (i.e., settlement finality - Directive 96/28/EC).

[75] In this writing we are using as reference the characteristics of the Algorand blockchain – a third generation blockchain – which we have specifically used since 2022 to continue, on a public blockchain, the development of contracts-on-chain which were started in 2019 on a private blockchain (Hyperledger).

[76] The degree of decentralization of public blockchains is relevant for several purposes. Firstly, it indicates the limits and thresholds for the validation of blocks that are to be controlled. The higher the number of validators, the more difficult it is for an attacker to take control of the validation system. This is important because control over who decides on block validation allows one to establish whether some operations inserted in a block are invalid. Secondly, the degree of decentralization affects the governance of the blockchain: the greater the number of validators, the higher the probability that they will participate in decisions on fundamental aspects, including development of the blockchain. The security level of blockchains has grown together with that of the generations of blockchains and has undergone an evolution that is partly independent and partly derived from the development of other characteristics of blockchains. As for the first element, some third generation blockchains have security protocols that use advanced, quantum resistant cryptography. As for the second, the growth in the degree of decentralization and scalability makes it objectively more difficult to compromise a block in the chain.

which the contract might risk being essentially a self-regulating mechanism.[77] This aspect is particularly noteworthy given the recent consolidation of Web2 logic platforms (i.e. social networks), which are becoming a sort of private legal order.[78]

To address these questions, we have developed contracts written in natural language on a public blockchain, taking advantage of the capabilities offered by third-generation blockchains. Since this is a public blockchain with transparent transactions, we have chosen to use it to enhance certain elements of contractual certainty without placing the entire contract on the blockchain, thus safeguarding contract flexibility as requested by the parties to the contract on a case-by-case basis. For instance, personal data and other identifying information of the parties are not disclosed on the public blockchain.[79] This choice does not affect the correct implementation of the contract, especially when, as explained, contracts-on-chain are embedded in a logical platform.

Contractual models within the financial sector were carefully examined and adapted to function seamlessly on a public blockchain and made receptive to the use of smart contracts.[80] This adaptation allows for the full harnessing of both the unique qualities that public blockchains offer as well as their potential contributions to any given contract, as we will detail below.

The intersection between natural language contracts and public blockchain – via software design, particularly smart contracts – has yielded surprising results. This merging appears to provide a secure and efficient approach to the current phase of Web3 development. In a short time span, the contracts-on-chain model could emerge as an innovative negotiation process complementing existing off-chain methods, bringing undeniable benefits in terms of security, transparency, and transaction speed. In support of this conclusion, we will proceed further by taking up the example of an escrow contract, something which presented – in the formulation of a smart legal contract (*supra*, Section 3) – severe challenges in terms of transparency. The contracts-on-chain model innovates significantly in this respect, increasing the usability and effectiveness of a smart legal contract tool that defines legal relationships between the parties according to "if-then" logic. In this case, in combination with an additional "abstraction layer", it offers the user a (party to the contract) an interface that facilitates interaction.

The novelty of contracts-on-chain is that they display all the contract clauses clearly, rather than just providing a representation of the contract in natural language while it is in fact written in machine language. In practice, once a user logs into the logical platform, she/he enters a system that not only offers a familiar user experience but is also remarkably comprehensible on a linguistic level. Contracts-on-chain are developed in a more complete and complex way than smart legal contracts. The technical/technological components are integrated in the initial phase of the development process as they are essential to creating a natural language contract that is immediately compatible with the Boolean logic of the blockchain.

This approach entails that the contractual terms are not simply agreed upon by the parties and then translated into programming language. Instead, they are collaboratively crafted from the outset by both technical and legal experts and meticulously designed to be executed automatically with the highest efficiency, taking into consideration the application environment. Consequently, "non-technical" users will rely on the natural language on the chain, as it perfectly aligns with the underlying functions of the smart contract.

For clarification, we provide the same section of the escrow contract as discussed in the section about smart legal contracts (Section3), consisting of mere lines of code, which has been constructed following the "contracts on chain" logic.

[Subject to Paragraph 4.2 of the Agreement, you are accordingly requested to pay the Deposited Amount, by [_] as follows:

(i) The Amount Deposited into the Deposit Account, totaling Euro [], is to be remitted to bank account no [_], IBAN [_], registered in the name of the Beneficiary and held at [_];
(ii) A portion of the Deposited Amount, amounting to Euro [_], designated as a Deposit Fee, should be transmitted to a bank account of your choice under your name.

Upon fulfilling the aforementioned payments, you are also requested to provide us with a confirmation ENCLOSING EVIDENCE OF THE TRANSACTION AND A STATEMENT OF THE DEPOSIT ACCOUNT CERTIFYING THE UPDATED DEPOSITED AMOUNT, following the provisions of Paragraph 4.5 of the Agreement.

[Beneficiary]
Name: [_]
Title: [_]
[Depositor]

In the contracts-on-chain version, this clause allows the user to engage in direct contractual negotiations on the blockchain and actively and consciously participate in all phases following stipulation. The platform provides users with the necessary tools to customize and determine the execution flow of the contract. The user-friendly interface incorporates elements (such as special 'buttons', akin to those found in common applications) that correspond to activities that can be performed based on the terms established during negotiations. For instance, parties can authorize the release of the deposited amount if they believe the agreed-upon activity has been satisfactorily carried out, or they can suspend the execution, transferring any flawed phases off-chain (see Fig. 1).

All in all, with contracts-on-chain, users have access to a technology that enables them to operate independently, without the intervention of specialized third parties, while also benefiting from the advantages

[77] From a common law perspective, the contract is the unique source of obligations existing between the parties. Smart legal contracts could therefore, albeit with the limitations highlighted above, represent the entire perimeter of the constraints between the parties. From the contract norm, however, there is a risk of moving towards a contract-legal order via blockchain, for those who believe that the blockchain constitutes a legal system in itself. According to the theories that transform *the rule of reason* into *the rule of code*, the blockchain possesses all the tools to ensure correct execution of the contract and allows for dispute prevention and resolution. It follows that a smart legal contract does not necessarily need to anchor the contractual rule to a third-party, a state law, or a given jurisdiction. The phenomenon, however, is neither new nor attributable to blockchain. Already in Web2, in fact, the relationships between digital platforms and users had often been unmoored from national jurisdictions.

[78] On this point see F. BASSAN, *Digital Platforms and Global Law,* 2021, cit. Some digital platforms (especially the 'closed' ones, such as social networks) operate with users on the basis of rules that the platforms themselves have defined (for example, with regard to admissible written or visual content) and which they are able to execute effectively (by obscuring accounts, if necessary), with these platforms also providing very articulated systems of dispute resolution. As for the relationship between digital platforms and their users, public systems (e.g. the European Union with the Digital Services Act) have recognized the independence of platforms, requiring compliance with general principles and, in fact, collaboration between the two legal systems (the private one of digital platforms and the public one of the European Union).

[79] On the public blockchain, only a recorded hash (#) identifies the contract in the state in which it is sent to the chain. On the one hand, this allows the parties to prove (and the platform operator to verify) that the contract identified with that hash is indeed the one in their possession. On the other hand, the contract is not recorded on the blockchain, so that it remains available only to the parties, who will be able to store it on premise or use one of the available cloud storage systems. Each modification of the contract will result in a different version thereof, which will be recorded and identified on the blockchain with a different hash. It follows that no personal data is recorded on the public blockchain, nor are elements that allow third-parties other than the parties to the contract and the platform operator to directly identify the parties.

[80] This involved: (i) a first phase of contractual simplification, so as to make it amenable to both translation into machine language and execution according to binary logic, and (ii) a second phase involving the transformation of some contractual clauses to make them self-executing upon the occurrence of certain events (on-chain or off-chain).

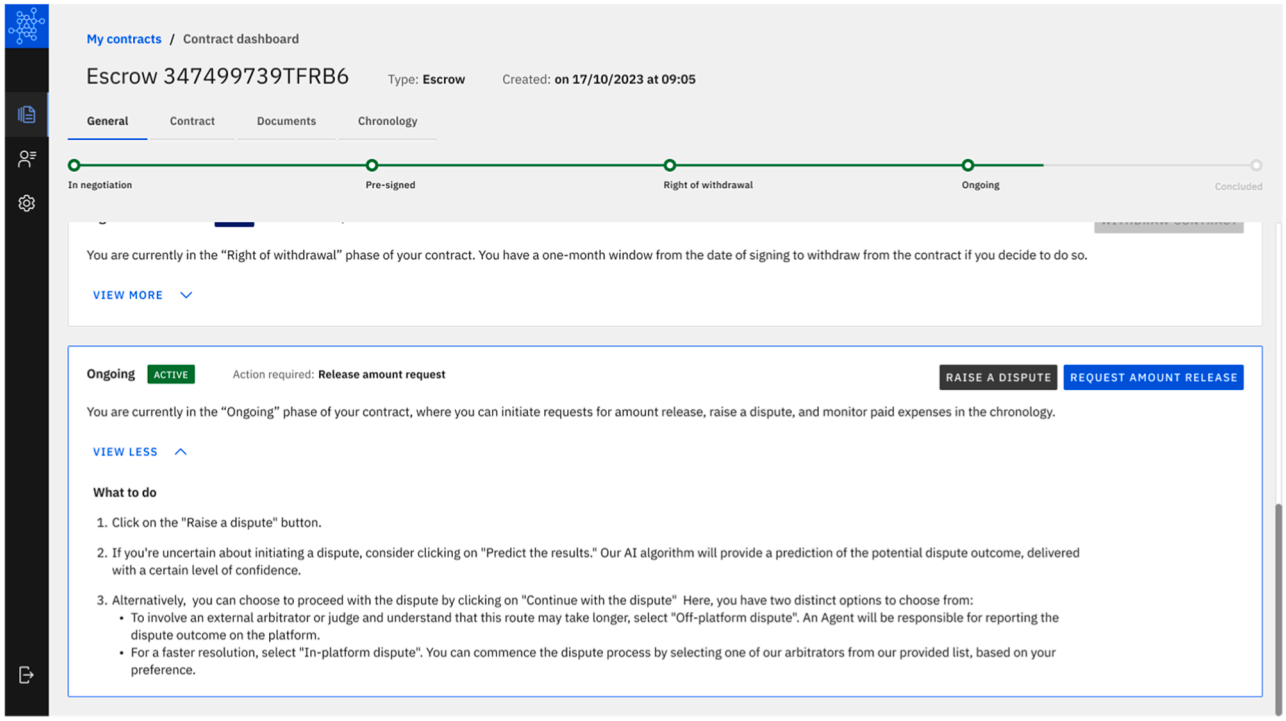


Fig. 1. Contracs-on-chain dashboard.

offered by the *public* blockchain as further enhanced by the safeguards provided by the logical platform.

Contracts-on-chain enable the alignment of every clause in the natural language contract with a corresponding logical clause within the smart contract. This alignment is achieved through the establishment of a direct 1–1 match (mapping) between the natural language clause and the smart contract clause. The following section of code pertains to the button labeled 'Request release of the deposited amount', as indicated in Fig. 1. This button allows users to execute the relevant function with a single click.

```
const handleProposeRelease = () => {
const rpcParams = {
fn: 'propose_release',
args: { msg},
onSuccessMessage: "Request to release the Amount
Filed completed",
modalConfirmMessage:
"You are about to request the release of the Deposited
Amount,
Do you want to confirm your choice?",
};
setRpcParams(rpcParams);
};
```

Since the code is readable and openly available on the public blockchain and can be verified with any blockchain exploration tool (explorer), the matching between the contractual clause in natural language and its corresponding logical clause within the smart contract, which constitutes its translation and execution, is effortlessly confirmable. This verification can be conducted not only by the involved parties but also by an arbitrator or a judge. Furthermore, if the subject matter falls under the oversight of a sectoral authority, that authority can also verify it, either through conventional methods like inspection by an expert or, conceivably, by implementing *Suptech* procedures.[81]

Hence, the distinctions between 'Ricardian contracts' (including hybrid contracts) and contracts-on-chain become quite evident, along with the novel elements introduced by the latter. 'Ricardian contracts' are essentially a form of software design; they are neither true contracts nor split contracts.[82] They were first developed by Grigg in 1996, before the rise of blockchain technology, making them independent of it.

[81] In this regard, and specifically for the SupTech evolution, see M. Rabitti - A. Sciarrone Alibrandi, *RegTech and SupTech*, Artificial Intelligence and Law: A Revolution?, edited by A. Pajno, F. Donati A. Perrucci, V. III, *Intellectual Property, Society and Finance*, 2022, pp. 451ff.

[82] It is therefore worth clarifying that we do not intend to refer here to the 'Ricardian contract' nor the 'split contract'. The Ricardian contract puts a legal agreement in a format that can be expressed and executed in software, including making it machine-readable and ordinary text-readable. The components of a Ricardian contract are disaggregated as follows: 1. a contract is offered by an issuer to contract holders; 2. It is held for a valuable right by holders and managed by the issuer; 3. It is easily readable on paper and by programs; 4. It is digitally signed, carrying the keys and server information, and is allied with a unique, secure identifier (hash). Thus, the 'Ricardian contract' is not a contract model but a software design, created in 1996 by I. Grigg to record the parties' intention to reach an agreement and to connect this representation of the will to third-party systems (i.e. accounting software) so as to give it legal effects. Recently, among external systems, it has become possible to use blockchain. This allowed the creation of a new version of the design software which is known as a Ricardian contract (a so-called 'hybrid' version) characterized by three elements: prose, parameters, code (Ricardian Triple): I. Grigg, *The Ricardian Contract*, in *Proceedings of the First IEEE Workshop on Electronic Contracting*, 2004, and Id. *Why the Ricardian Contract Came About? A Retrospective Dialogue with Lawyers*, in J . G. Allen - P. Hunn (Editors), *Smart Legal Contracts: Computable Law in Theory and Practice*, Oxford, 2022, p. 88 e ss. And, equally, we are by no means referring to the 'split contracting model', which, unlike the 'Ricardian contract', immutably connects the contract in written form, understandable by humans, with the architecture – which remains separate – of smart contracts, understandable by the machine that administers its execution.

Recently, the integration of smart contracts with Ricardian contracts has allowed them to evolve in line with blockchain, although they still face limitations due to their inherent rigidity (the contract is identified by a single hash).

Conversely, contracts-on-chain are real contracts that shift the concept of smart contracts, including smart legal contracts, from the realm of theory and technology into potentially widespread practice. Contracts-on-chain exist solely within the blockchain, making it an essential component. In this practice, the contracting parties utilize tools, methods, and models widely established and familiar from online contracts, complemented by the elements inherited from the blockchain on which they operate.

Contracts-on-chain take advantage of the recent evolution of third-generation blockchains (Section 5. *i)*), which are scalable, economical, and energy-efficient. This allows every exchange between parties to be recorded on a public blockchain, including negotiations, signatures, contract modifications, execution (whether partial or full self-execution), also using native blockchain tools for resolving disputes. Each step can be recorded on the blockchain based on the requirements set by legislators, regulators, or the preferences of the parties involved.

The flexibility of contracts-on-chain ensures that parties have complete 'internal control' of the tool. This flexibility is further enhanced by integration with logical platforms (Section 5 *i)*) that enable 'external control' through access requirements to the platform.

The next section will delve into how this relates to the specific characteristics of these contracts.

6. The features of contracts-on-chain: the premise

The possibility of verifying linguistic correspondence between computer code and natural language is undeniably a crucial element in ensuring respect for the parties' intentions. To emphasize this point, we further explore the impact of this novel approach to drafting, concluding, and executing contracts – which we refer to as 'contracts-on-chain' – on the issues raised by legal scholars regarding the concept of smart legal contracts. In particular, it bears examining the issue of translation risk (language), the profile of (in)flexibility, and the applicability to smart contracts and to smart legal contracts of safeguards provided by traditional contract law, as well as the applicability of traditional conflict-of-laws rules.

(i) Translation risk

It is commonly asserted that a seemingly insurmountable limitation of smart legal contracts, which are typically written in a manner comprehensible only to engineers, is their potential failure to meet the requirements of clarity, comprehensibility of contractual texts, and, in the financial sector, information transparency.

One tangible risk arising from this scenario is that smart legal contracts may provide and execute in a manner different from the intentions expressed by the parties in the natural language contract. In such a case, the automatic execution inherent in smart legal contracts might give priority to the code written by the person who authored the smart contract over the intentions of the contracting parties. We can consider a scenario where the parties agree on a purchase price for an asset via blockchain and the smart legal contract's translation inadvertently alters the amount to zero or to another unintended value. When the predetermined condition is met, the automatic execution results in an incorrect outcome. This represents a potential error not by the parties, but rather by the "translator" – an error that remains hidden from the parties' view.

Nonetheless, this issue can be resolved by applying the contracts-on-chain model within a logical platform that operates on the latest generation of public blockchains. These platforms are *open source* and transparent and allow anyone authorized by the parties (such as platform operators) to verify the consistency between the two versions of the contract, with even basic coding knowledge being sufficient for such an inquiry.

An alternative solution might involve incorporating mechanisms for both *ex-ante* and *ex-post* verification. For example, parties could have the option of requesting certification of the correspondence between text signed off-chain and its encoded counterpart. This certification could be obtained from a trusted third party, hypothetically the logical platform operator. Alternatively, an oracle, whether operated by the platform itself or by a third party mutually agreed upon by the parties, could be charged with validating the consistency between the natural language and machine language text on the blockchain. This validation process could result in a compliance certificate, which would then be recorded on the blockchain, and thereby carry significant evidentiary effects.

This solution, aimed at facilitating consistency checks between the first and the latest (negotiated) version of the contract, can also beneficially address the broader issue of non-IT-savvy contractors not being aware of the terms and conditions of the concluded agreement. In this way, it would effectively eliminate "translation errors" at their source, allowing parties to enjoy the advantages of placing contracts on the *public blockchain* without exposing themselves to the associated risks.

Furthermore, it is worth noting that the effort required to clarify and simplify the language, essential for converting contractual clauses into code in an on-chain contract, may, in some instances, introduce limitations on the degree of execution flexibility. Nonetheless, this approach can help to overcome the inherent ambiguity often associated with contractual terminology.[83]

ii) The inflexibility of smart legal contracts

The second paramount concern, as previously mentioned, revolves around the (in)flexibility inherent in smart legal contracts, especially when a need to renegotiate has arisen as a result of unforeseen circumstances, integration requirements, a party's withdrawal from the contract, or exceptional or "pathological" circumstances necessitating contract interruption. It is generally acknowledged that smart legal contracts are a means to transcribe the terms and conditions agreed upon by the parties into code and store them within the blockchain, thereby rendering them verifiable, immutable, and irrevocable. From this perspective, the rigid nature of smart legal contracts raises questions about their capacity to accommodate and apply rules related to contractual deficiencies, including nullity and termination.

The most advanced blockchains now offer a certain degree of adaptability and revocability. They permit modifications of smart legal contracts – even during their formation – if the option to revoke, modify, or cancel is embedded in the code from the outset. This means that smart contracts still provide a measure of flexibility. Contracts-on-chain, however, offer an additional feature: they allow the parties, in response to expressed needs (including needs articulated by a single party that have been deemed legitimate by the counterparty, an oracle, or a judge – even off-chain), to modify an already concluded agreement and replace the old smart contract with a new one that can be negotiated directly on the blockchain.

[83] Many believe that it is precisely the vagueness of clauses such as 'best efforts' or 'good faith' that make it difficult to utilize smart contracts in complex contractual arrangements. See, among others: C. Poncibò – L. Di Matteo, *Smart Contracts Contractual and Noncontractual Remedies,* The Cambridge Handbook of smart contracts, blockchain technology and digital platforms*, (L. DiMatteo et al, eds., 2020), p. 120-121; p.* Catchlove, *Smart Contracts: A New Era of Contract Use, 2021*, available on https://ssrn.com/abstract=3090226; M. Rudanko, *Smart Contracs and Additional Contracts: Views of Contract Law*, Smart Contracts*: technological, business and legal perspectives,* M. Compagnucci et al. eds., 2021, pp. 59 ff.; L. Di Matteo, *Smart Contracts and Contract Law,* The Cambridge Handbook of smart contracts, blockchain technology and digital platforms*,* (L. Di Matteo et al, eds., 2020), pp. 8ff.. It will be seen, below, how contracts-on-chain can overcome these limitations (Section 14, dispute resolution).

At the same time, the operation remains recorded on the public blockchain, which documents the entire sequence, including all changes made during the lifecycle of the contract. In this way, contracts-on-chain transform a fundamental limitation of smart legal contracts (rigidity) into a tool (irrefutable registration) that can be employed, especially in the event of "pathological" scenarios.

(iii) the applicability to smart contracts and to smart legal contracts of safeguards provided by traditional contract law,

As we mentioned, for public blockchains, the issue of potential anonymity does arise. To address this challenge, we propose the use of logical platforms (see section V). We believe that using logical platforms on a public blockchain is the best way to showcase the potential of contracts-on-chain. Access to these logical platforms requires successful completion of various procedures, including participant identity verification, KYC procedures, and AML regulations. This ensures that a public blockchain can offer the same safeguards as a private blockchain, while still benefiting from the superior security, decentralization and scalability of a public blockchain. Essentially, the logical platform mitigates the risks associated with a public blockchain while providing the guarantees of a private blockchain. Furthermore, the modular use of blockchain for purposes such as identity verification, negotiation, agreement, execution, and dispute resolution—whether required by legislation, regulators, or the parties involved—makes contracts-on-chain a highly flexible tool. Contract law applies fully to contracts-on-chain because, in parts where blockchain is not used, traditional legal principles apply, and in parts where blockchain is used, it ensures all legal guarantees are met. As we will demonstrate, using blockchain can actually enhance these guarantees through regulation by technology.

Thus, the combination of the Algorand infrastructure (a third-generation public blockchain) and the logical platform ensures that contracts-on-chain can be fully recognized and governed by contract law.

(iv) the applicability to smart contracts and to smart legal contracts of traditional conflict-of-laws rules

The combination of the 3rd generation public blockchain with the logical platform and the contracts-on-chain negotiation tool also allows for the full application of the current tools of Private International Law. The connecting factors (party autonomy, the law of the closest connection, the lex loci contractus, and the lex rei sitae) can be applied to contracts-on-chain.

The lex loci solutionis would apply as well, for the place of performance can be always identified. This helps resolve uncertainties about the location of the object when it is inherently digital.

Similarly, overriding mandatory rules can be applied to contracts-on-chain.

In the following sections, we will explore how contracts-on-chain, operating on a logic platform, on the one hand address questions raised by legal scholars regarding the compatibility of contractual rules and smart legal contracts and, on the other hand, help to reduce – and under certain profiles, eliminate – issues related to translation risk, the inflexibility of smart contracts, the applicability to smart contracts and to smart legal contracts both of safeguards provided by traditional contract law, and of traditional conflict-of-laws rules; all aspects which have so far limited the use of blockchain for contractual purposes.

We will rely on the model presented by the Italian Civil Code, which serves as a civil law framework, primarily focusing on the aspects of agreement, formation of consent, object, and form.

7. Identification and capacity to act

The first issue relates to the need to identify the contracting parties and to ascertain whether they are competent to enter into agreements.

The identification of a user on a public chain is typically done with a wallet, mainly to carry out transactions. Identifying the parties to a contract on the blockchain requires additional steps, which a logical platform on the public chain can guarantee.[84] Some scholars who have studied smart legal contracts have raised concerns about the difficulty of identifying and confirming the legal capacity and competence of participants, particularly in cases where certain blockchains potentially allow anonymous, pseudonymous, or even untraceable actions.[85] While this challenge might theoretically exist in the context of decentralized public blockchains, it is likely to be mitigated or resolved when employing the contracts-on-chain model within a logical platform. In this model, the entity operating the platform can, and often must, authenticate the identity and legal capacity of the involved parties.[86] Regardless of whether a logical platform is in use, contemporary electronic and digital identification tools, as well as native blockchain solutions, can effectively address these concerns.

The advantage of using the public blockchain is rooted in the enduring nature of identification records. Once an identity has been established and verified, it remains permanently documented on the blockchain. Even when concealed by hashes, a manifestation of intent or a transaction can always be traced back to a specific natural or legal person. While it is possible to modify identification for future contracts undertaken by the same party, the public blockchain keeps records of the past identifications. This persistence occurs because the identification process is carried out via smart contracts and thus inherits their characteristics, as discussed earlier (see Section 6).

8. Contract formation: negotiations

As previously mentioned, third-generation blockchains offer the advantage of low transaction costs and almost instantaneous execution speed,[87] which allows each step of the negotiation and contract formation process to be recorded in an efficient, cost-effective, permanent, and immutable manner.[88]

From this perspective, the possibility of negotiating the content of the contract directly on the blockchain is also important in assessing the

[84] *Supra*, Section 5.

[85] Public blockchains are characterized by the publicity and transparency of transactions, which take place in pseudonymous form. The addresses of the transferor and transferee are public, unlike the identity of the subjects who control the private keys of the crypto-assets associated with them. Moreover, there exist some technological solutions that guarantee anonymity, especially for the purposes of circulating virtual currencies. For example, privacy coins (Monero, Zcash, Dcash), mixers, tumblers and other tools that obfuscate the traceability of transactions. Obviously, many of these instruments raise concerns, as they do not comply with the current regulation on transparency and, for example, AML. By contrast, the Contracts-on-Chain model inserted within a logical platform, moves exactly in the direction imposed by current legislation.

[86] Obviously, the role played by the operator of the logical platform can also more broadly encompass all the functions of a Qualified Trust Service Provider (QTSP – according to eIDAS semantics) which – owing to training, audit burdens and accreditation – can also be fundamental in the decentralized world when it comes to guaranteeing: a) the identity of the parties; b) storage of the cryptographic keys used in transactions; c) security, and d) an understanding of the contents.

[87] The relevance of transaction costs is thorough analyzed by A. Delgado De Molina Rius, *Smart Contracts: Taxonomy, Transaction Costs, and Design Trade-offs*. In: Smart Legal Contracts. Edited by: Jason Grant Allen and Peter Hunn, Oxford, 2022.

[88] *Supra*, Section 6.

exact performance of the parties.[89]

More generally, the concept of 'notarization' during the negotiation phase holds significant importance, especially in a civil law jurisdiction, where the principle of good faith in contracts permeates all aspects of the parties' conduct and judicial reasoning. Under civil law, the obligation to act in good faith during negotiations, aligning one's conduct with principles of loyalty and fairness, entails pre-contractual liability in the event of rule violations.

Under the framework that we have outlined, involving a logical platform operating on a 3rd generation public blockchain, the various stages of contract formation are meticulously documented and made accessible to a judge or arbitrator as necessary. These legal professionals are to review the contractual agreement and scrutinize the conduct of the parties throughout the contract's entire lifecycle. This detailed record of events, firmly entrenched within the blockchain, stands as secure and incontrovertible evidence of the contractual history. This multifaceted capability – documenting the negotiation and pinpointing accountability, particularly as might relate to a breach of the good faith principle – greatly facilitates the negotiation process

An illustrative example pertains to a breach of the obligation to provide information, notably in cases involving specific pre-contractual or contractual disclosure requirements within domains such as banking, finance, and consumer contracts. Access to the blockchain ecosystem often furnishes ample evidence to ascertain the proper fulfillment of these obligations. The potential integration of contracts-on-chain within a logical platform serves to notably alleviate the burden of proof, thus streamlining the assessment process.

For instance, in the context of an investment contract, contracts-on-chain have the potential to ensure adherence to regulatory requirements on disclosure and regarding the conduct of financial intermediaries in the realm of investment contracts, as stipulated by MiFID 2 and the investor protection regulations delineated in the Italian TUF (Consolidated Law on Financial Intermediation). If parties opt to embrace the contracts-on-chain model, the verification of whether prerequisite information has been accurately conveyed, and whether it has been presented in a clear and comprehensible manner, is facilitated, given that this information is securely logged on the blockchain complete with a timestamp. Moreover, if a contract clause adhering to legal obligations is subsequently integrated into a smart contract within the domain of contracts-on-chain, it will autonomously execute its obligations in conformity with the law.

From an alternative perspective, the blockchain serves as a valuable tool in identifying potential unfair clauses within consumer contracts. These terms have no legal or binding force on consumers because they are void, due to European law. Blockchain enables the elimination of unfair terms through a designated kill function when implemented within smart contracts or by way of a declaration by a judge or oracle.

9. The process of concluding contracts-on-chain. Agreement and form

Through the adoption of contracts-on-chain, the parties elect to employ a distinctive procedure for contract conclusion. This choice is not predicated on a deviation from the conventional proposal and acceptance mechanism but rather arises from its execution on the public blockchain, which gives rise to a novel negotiation mechanism as an expression of private autonomy. In certain regards, this mechanism can be expected to have an impact/a positive impact on the legal consequences for the parties involved.

However, the technical characteristics of public blockchain and smart legal contracts lead us to emphasize once again that not every contract can benefit from stipulation and execution on this infrastructure. This platform is most advantageous and effective for contracts that place a premium on notarization, for contracts containing stipulations that are relatively immutable, and for contracts featuring contractual clauses that have the ability to self-enforce. It is within this context that the contracts-on-chain process endows these features upon contracts drafted in natural language on the blockchain, thus affording parties the ability to combine the advantages of traditional/digital contracts with those of smart legal contracts.

Indeed, contracts-on-chain can leverage the rigidity of the blockchain to ensure the certainty and security of activities that are notarized, while at the same time giving parties the flexibility they deem appropriate, thus, for example, allowing them to modify the contract when certain conditions occur.

Contracts-on-chain exhibit remarkable efficiency, especially in scenarios involving business contracts, such as escrow arrangements, as previously explored. They are also highly suitable for standardized contracts encompassing various domains like investments, financing, insurance, and consumer credit; their adaptability to conventions and framework agreements should also be stressed. Furthermore, they serve well in specific consumer contracts, notably within regulated sectors such as communications, energy, water, and waste management.

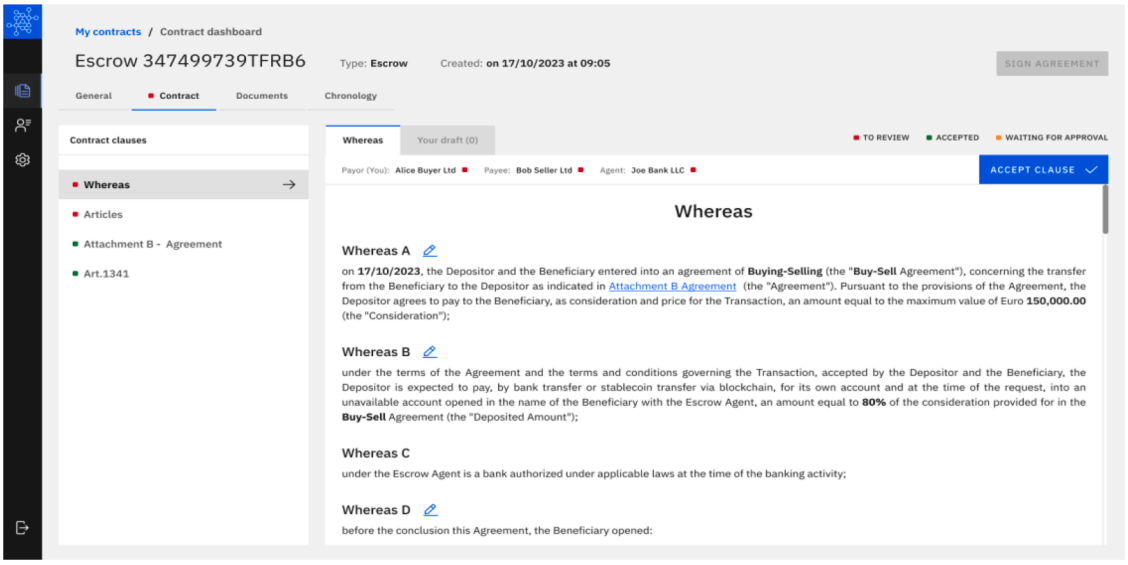
The value of smart legal contracts in the form of contracts on chain is that they seamlessly facilitate/further a process whereby the initial negotiation phase is conducted and completed directly on the blockchain. When the parties have successfully reached an agreement, the contract requests the endorsement by signature of both parties for valid and partial execution. This endorsement can be accomplished with common tools designed for advanced or qualified digital signatures, or even via native blockchain signature tools. As with digitally signed contracts, these signatures possess a robust level of authenticity that is difficult to disprove except in the case of identity theft.

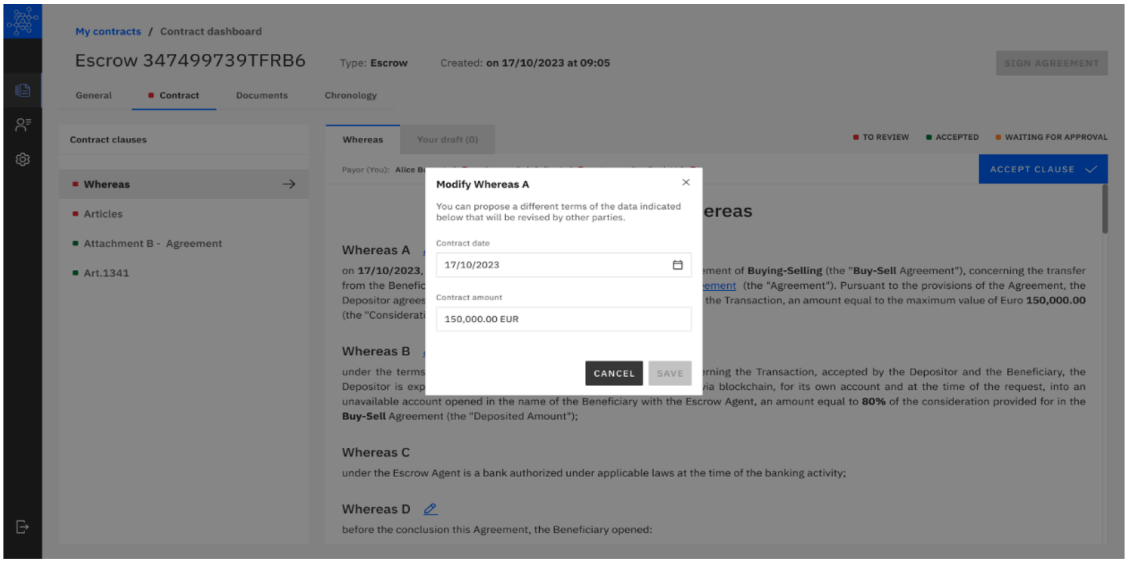
In third-generation blockchains, characterized by high speed and exceptionally low costs, individual clauses are proposed and accepted directly on the blockchain after having been initially drafted in natural language and gradually inscribed onto the blockchain using smart contracts that mirror the content of the agreed-upon clauses. Consequently, these smart contracts execute the contract in compliance with the terms agreed upon by the parties. The contract, which has been both negotiated and concluded directly on-chain, undergoes full recording throughout its phases on the public blockchain, thereby preserving a comprehensive record of all contractual negotiations, including the proposal and acceptance. Within the blockchain environment, the text of the negotiations and the final signed contract are available in PDF format, shielded by a hash and encryption that ensures the security and integrity of the enclosed data. Once finalized (concluded) and when the specified conditions are met, the contract will be self-executing, i.e. it automatically enforces the agreed-upon terms as established by the contracting parties.

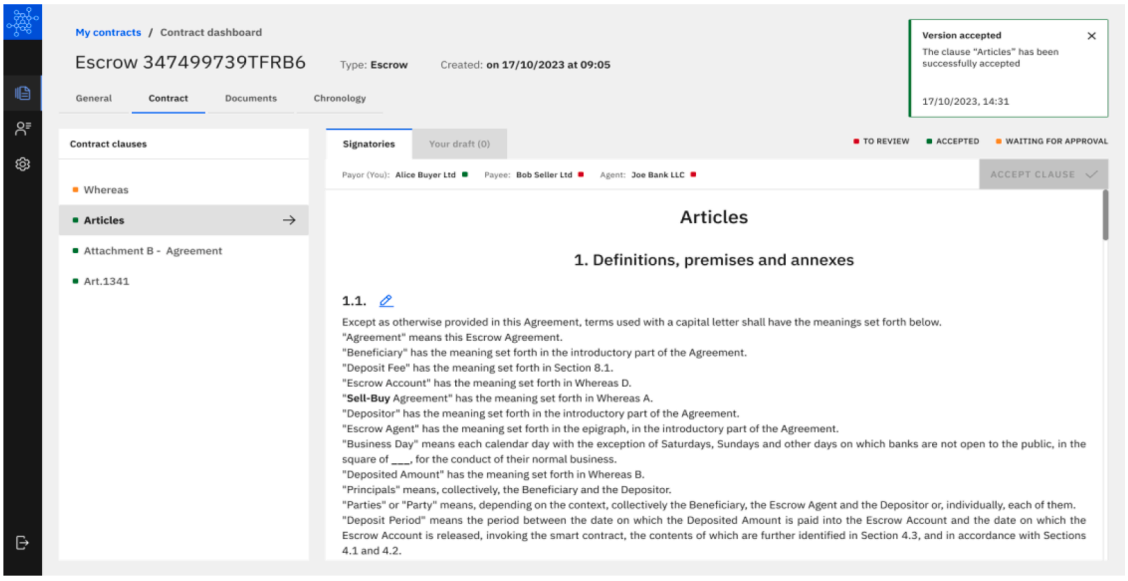
In the context of the framework outlined herein, it is noteworthy that the academic debate on smart legal contracts, devoid of the contract-on-chain structure, tends to conceptualize smart legal contracts as being akin to contracts that are concluded at the inception of execution. To describe how contract negotiation and conclusion would take place within the contracts-on-chain paradigm and to identify the practicalities, we shall consider the example of an escrow contract. Specifically, we will illustrate the evolution of negotiations occurring between the involved parties concerning the deposited amount. The inherent capability of contracts-on-chain facilitates the individual participation of the parties, allowing them to directly enter or modify the desired amount within the text of the contract (see the figures, below). Upon attaining through this mechanism a consensus on the terms subject to negotiation, the contract is signed and executed on-chain, achieving full operational

[89] See: M. Durovic-A. Janssen, *The Formation of Blockchain-Based Smart Contracts in the Light of Contract Law,* (2018) 26 European Review of Private Law 753; M. Cannarsa, *Interpretation of Contracts and Smart Contracts: Smart Interpretation or Interpretation of Smart Contracts?* (2018) 26 European Review of Private Law 773; R. Caria, *The Legal Meaning of Smart Contracts*, (2018) 26 European Review of Private Law 731.

status and binding both parties in its definitive version.







The distinction between contracts-on-chain and smart legal contracts is clear-cut. Smart legal contracts, due to their purely coded nature, are not comprehensible to the parties involved. As such, they do not facilitate direct on-chain negotiation, regardless of whether or not they are drafted according to a hybrid model. It is, furthermore, worth highlighting that the use of a hybrid model does not eliminate the risk of language discrepancy. By contrast, contracts-on-chain represented in a code language may appear as follows:

```
updateagreementdraft:
Proto 6 0
callsub stateisnegotiation
// Agreement draft must be in Negotiation (WRONG_STATE)
Assert
txn Sender
callsub isescrowagent
// Caller must be the Escrow Agent (WRONG_SENDER)
Assert
[...]
frame_dig -4
intc_0 // 0
>
// Amount must be greater than 0 (WRONG_AMOUNT)
Assert
[...]
bytec_3 // "amount"
frame_dig -4
app_global_put
```

10. Probative value

The possibility of retracing a document to a specific person depends on the recognition system. The strength of the security guarantee depends on the type of signature used, which determines different legal effects. Specifically, in terms of reliability and security, the strength of the signature lies: (i) in the complexity of the signatory identity verification system and (ii) in the possibility of proving that the document corresponds to the one that was signed. Based on these criteria, the eIDAS Regulation (EU Regulation 910/2014 on electronic identification and trust services for electronic transactions in the internal market) recognizes the 'qualified electronic signature' (QES) – and not the already existing 'simple electronic signature' (FE) or 'advanced electronic signature' (FEA) – as the instrument that guarantees the highest level of security in terms of authenticity, integrity, and non-repudiation. The QES makes use of cryptography and provides for the use of a qualified certificate. From a legal point of view, a qualified electronic signature is, for all intents and purposes, equated to a handwritten signature. Therefore, it will be up to those who challenge its validity to prove the non-existence of the signature.

Furthermore, as concerns the interests being safeguarded and promoted, from a functional perspective the emphasis lies on the objective of protection (the end) rather than on the technical solution (the means), with the latter relating to the traceability of the document to its author and the suitability of the chosen mechanism to ensure the security, integrity, and immutability of the document. An evolutionary interpretation of digital identification in this direction is appropriate, also in view of the rapid pace of technological solutions.

11. Negotiation link

Contracts are initiated, undergo negotiation, are concluded and executed, often have different parties, follow different models and rules, and are subject to different jurisdictions.[90] A contractual linkage – a fundamental construct/institution for complex contractual transactions – comes into existence when autonomous contracts, each with their own purpose (*causa*), become part of an overarching operation aimed at realizing an additional purpose separate from that associated with the individual contractual purposes of each linked contract.

As a result, the functional connection between contracts is significant in that they must be regarded in respect of the practical outcome they collectively aim to attain, given that the overall function binds the entire transaction. This first objective requirement is coupled with a subjective one that entails the shared practical intent of the parties, who desire not only the customary consequences of the specific transactions but also an additional purpose. It follows that: (i) to assess the lawfulness of the economic transaction undertaken, regard must be given to the overall function/purpose of the transaction, whereby the identity of the parties is not required; (ii) when interpreting the contract, account must be

[90] This choice is made either according the will of the parties, when expressed, or in the absence thereof, in light of the connecting factors established under private international law.

taken of the priority of interests resulting from all the linked contracts; (iii) when assessing the merits, lawfulness, and validity of the transaction, it is necessary to look at the set of interests concretely established by the parties.

Contracts-on-chain guarantee a genetic and functional contractual link that is automatic and permanent. It is sufficient to use the public blockchain to record (with a timestamp) the link between two or more contracts, thereby ensuring that each contract becomes effective and enforced only when the other contracts do so at the same time. The use of natural language, which characterizes contracts-on-chain, allows the parties to the different contracts to easily review the content of the agreements and directly monitor the progress of multiple contractual negotiations, which are all linked and accessible from their dashboard. The link between the contracts will also be partially visible (limited here to self-executing clauses) on the blockchain, as the conditions provided for by one contract will be automatically incorporated into another linked contract, forming a condition for its execution.

This link, once recorded in the blockchain, remains permanent, as does the execution via smart contract. This principle holds even in the case of complex relationships involving multiple parties. Suppose that a contract of sale is linked with (i) a contract for the transport of goods, (ii) an insurance contract, (iii) guarantees (for both parties, iv) possibly also an escrow contract, and (v) financing for the buyer.

In this case, the public blockchain appears to offer the means for ensuring that these contracts can be negotiated, signed, and executed on the blockchain only where they are part of a unified block. In other words, the negotiation of each contract will be documented on chain, and the signature of the parties to a contract can deemed valid only where it enters the chain at the same time as the signatures of the other parties to all the other linked contracts. All signatures are recorded within the same block.

Contracts-on-chain serve the purpose of providing an effective compliance tool. For instance, consider an investment contract designed to execute only after a positive verification of the completion of the MiFID questionnaire by the investor within the applicable deadlines (with such verification also subject to registration on the blockchain) and after establishing the suitability of the investor to enter the contract, as based on the outcome of the questionnaire. All of this, recorded on the public blockchain, remains permanently verifiable, as the two actions are intertwined to ensure the highest level of compliance with regulations.

12. Execution

One of the advantages of public blockchain is that the contract can be partly executed in an automatic, certain, transparent, secure, cost-effective, and immutable manner (*supra*, para 6). The automatic nature of the execution may be conditioned upon certain variables, agreed by the parties (e.g. the occurrence of particular events). Some of these conditions are implemented directly on the blockchain, others off-chain. As for the former, consider the example of purchasing a blockchain-based bond, which involves a blockchain-native token that circulates through smart contracts. In such cases, it is straightforward to monitor the bond's life cycle within the blockchain and verify the accuracy of automatic processes, such as changes in value and interest rates. However, when the conditions for contract execution rely on off-chain events, the parties can make use of public and objective references for validation. For instance, the determination of the interest rate of a loan agreement might be linked to a predefined variable (e.g. EURIBOR). In other cases, when the occurrence of the condition or event is not tied to an objective public reference, the parties may opt to have a trusted mechanism provide verification: the oracle. The oracle could periodically confirm that specific off-chain events have occurred in a sales contract, such as the delivery of goods to the carrier, successful customs clearance, and handovers to the purchaser without dispute. The oracle is responsible for recording these off-chain events on the blockchain.

Moreover, even the counter-performance (the non-characteristic service), if it consists of the payment of a price, can occur either on-chain (when the payment is made in crypto-assets,[91] in accord with the latest European definition, or cryptocurrencies[92]) or off-chain (if the payment is made in FIAT currency).[93]

When both performance and counter-performance occur on-chain, third-generation blockchains afford an 'atomic' exchange, which either takes place in the same block or does not occur at all. The technical revolution that blockchain unfolds, concerning not only real-world contractual affairs but also alternative solutions thereto, is crystal clear. It guarantees the immediacy and 'finality' of the transaction and reduces or eliminates the risk of default by the parties. Moreover, if integrated into a logical platform, the transaction does not entail a counterparty risk, because any operations on the platform presuppose the prior identification of parties, assets, and wallets.

The Proposed European Data Act also requires smart contract

[91] The well-established definition of crypto-assets can be found in the MiCA Regulation, which provides a harmonized regulatory framework for crypto-assets at European level. Art. 3(1)(2) of MiCA describes crypto-assets as "a digital representation of value or rights which may be transferred and stored electronically, using distributed ledger technology or similar technology". "Digital representation of values or rights" includes tokens (cryptocurrency, utility tokens, security tokens, non-fungible tokens). The Regulation, however, regulates only certain types of crypto-assets on the market: (i) asset-referenced tokens; (ii) e-money tokens; (iii) other crypto-assets such as utility tokens.

[92] Cryptocurrency is the most well-known type of crypto-asset; in addition to being used to trade products or services, cryptocurrencies can be a vehicle for speculation (such as trading on a cryptocurrency trading platform), or as a store of value.

[93] The counter-performance (the payment) on the blockchain can be carried out in different ways, all of which are currently being tested in the Eurozone by the ESCB (European System of Central Banks) on DLT systems integrated with Central Bank money. There are three main options that have been proposed, respectively, by the Bank of Italy (TIPS hashlink), the Banque de France (Central Bank Digital Currency), and the Deutsche Bank ("Trigger" solution). Trials will be launched between May and September 2024 following a call-for-interest that will identify and gather potential participants in the trial. These solutions must enable interoperability between infrastructures that are currently independent – both those that govern the exchange of digital assets and those that provide settlement services in central bank money. The aim is, in essence, to build a bridge between DLT and financial market infrastructures. The proposal from the Bank of Italy is described in detail in paper no. 26 of 2022, *Integration of DLTs with market infrastructures: analysis and proof of concept for a secure DvP between TIPS and DLT platforms* (available on the website of the Bank of Italy).

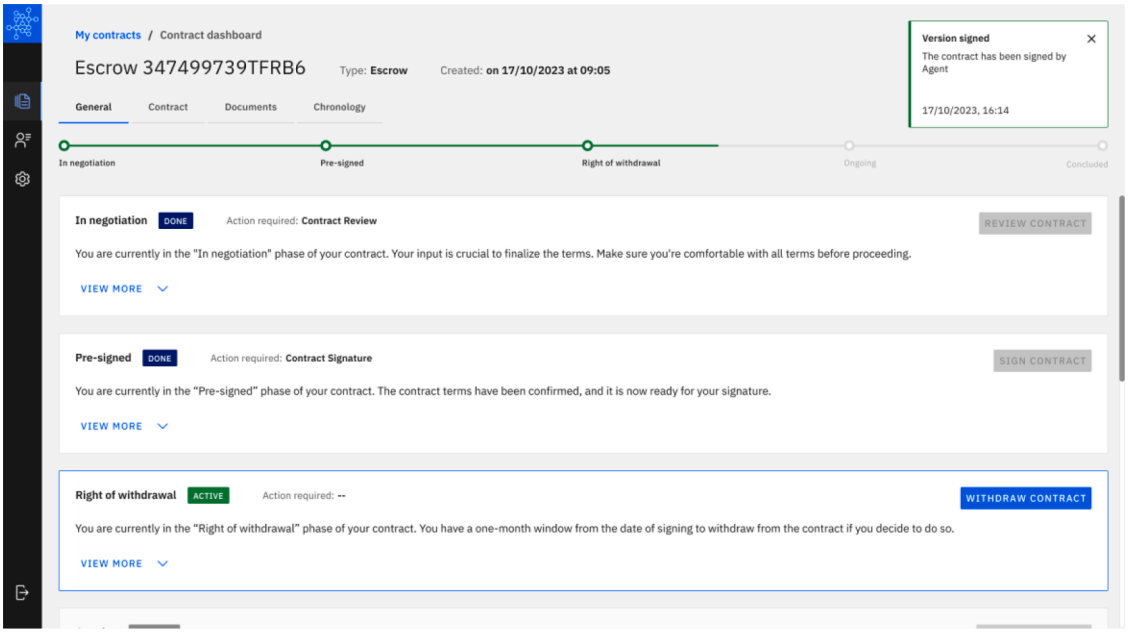
developers to ensure the possibility of halting contractual execution under specific conditions. This provision has sparked a heated debate among market operators and scholars. Some (e.g. Polygon) consider it not to be technology-neutral, as it would penalize public blockchains compared to private ones. The adoption of contracts-on-chain in the logical platform model that we have proposed (*supra*, Section 5) makes public blockchains compliant with the rules imposed in the current version of the Proposal for a Data Act concerning 'data sharing' and 'interruption'. Contracts-on-chain can already guarantee automatic interruption – either upon the occurrence of on-chain events or due to external intervention (e.g., an oracle) when the conditions occur off-chain and the parties have so arranged. It is therefore a matter of providing, alternatively, options for annulment by means of a 'reverse transaction', a code re-writing (re-coding), or the exercise of a 'kill function'. The kill function, in a specific environment, such as that of a logical platform on a public blockchain, may play a residual role.

For example, some scholars refer to the kill function as a guarantee of the exercise of the right of withdrawal in smart contracts. However, parties using contracts-on-chain can naturally exercise the right of withdrawal without the need to activate a kill function; such a process is more straightforward. Below is an example illustrating the right to exercise withdrawal within the escrow contract developed according to contracts-on-chain logic.

[For the duration of the Deposit Period, within 1 (one) Business Day of receipt of and in accordance with the Qualified Instructions, the Depositary shall return, in whole or in part, the Deposited Amount to the Depositor:

(i) who has exercised his/her right of reconsideration within [_][SG1] days [Insert number of days] of signing the Contract, pursuant to Article [6.3] of the Investment Conditions]

The party/user will be able to exercise the right of withdrawal in a simple and direct way on the blockchain by selecting the specific button provided by the logical platform.



If a party chooses to exercise the right of withdrawal, the smart contract will perform the following verifications: (i) it will confirm that the withdrawal request is made within the stipulated time limit, and (ii) it will ensure that the request is initiated by the depositor and not by any other party.

Upon successful completion of these checks, the smart contract will promptly and automatically refund the deposited amount to the depositor and notarize an on-chain timestamp to document this return. This marks the completion of the contract-on-chain operation. A comprehensive record of all transactions executed by the involved parties will stay on the blockchain, with access limited to the parties or individuals chosen by them, primarily for evidentiary purposes.

13. Modification of the contract

The European Data Act[94] requires, among the conditions that a smart contract must comply with, the possibilities of interruption, execution, and modification. At first glance, modification might seem ontologically incompatible with the inherent characteristics of smart contracts. Upon closer examination, it is not. In fact, smart contracts involve recording on the blockchain the intentions of the parties, the payment, or parts of the agreement designed for automatic execution. The blockchain used is not static (and indeed must evolve continuously) and neither are the smart contracts that operate on it. Contracts-on-chain, which are contracts that partly operate on the blockchain and that harness the characteristics of smart contracts, can be subject to modifications over time according to the intent of the parties or any external conditions. What is preserved on the blockchain is documentation of various contract versions (the first until the most recent), with the last version subject to changes by the parties in a way that resembles off-chain contractual practices. In this way, contracts on-chain merges the advantages offered by public blockchains with responsiveness to the need for flexibility typical of legal contracts. At least until now, legal scholars have deemed these conflicting needs hard to reconcile.

For example, in the face of an unexpected event that prompts the parties to renegotiate the contract - a scenario that has gained increasing relevance in recent years, especially following the financial crises and the Covid-19 pandemic - contracts-on-chain can guarantee the ability to revise certain clauses or sections of an on-chain contract. This can be achieved either by stipulating renegotiation triggers within the smart contract, by enabling the parties to invoke an oracle that communicates the necessary actions to the smart contracts, or, in the case of supervening and unforeseen events, by directly modifying the contract on the blockchain. Also, in cases where the parties cannot reach an agreement, they may resort to an oracle for assistance.

Parties involved in an on-chain contract could potentially delegate an oracle not only to report off-chain events on-chain[95] but also to determine performance or address aspects not initially covered by the contracting parties.

14. Dispute resolution

An sotware or hardware oracle[96] may have the task of resolving disputes.[97] This function can be performed by a trusted third party[98]: alternatively, by the blockchain technology provider or by the operator of the logical platform where contracts-on-chain are negotiated, concluded, and executed, or even by external entities chosen by the parties.[99] Contracts-on-chain ensures the flexibility that parties require,

[94] Regulation (EU) 2023/2854 of the European Parliament and of the Council of 13 December 2023 on harmonised rules on fair access to and use of data and amending Regulation (EU) 2017/2394 and Directive (EU) 2020/1828 (Data Act).

[95] Specifically, an oracle consists of a script that verifies the existence of the necessary prerequisites for the realization of the planned consequence. The data that oracles transmit drives the execution of smart contracts and thus the functioning of the blockchain ecosystem. For this reason, some call them 'bridges' between the real world and the blockchain protocols. See B. Carron - V. Botteron, *How smart can a contract be? , supra* n. 23, define oracles as "sensors in the physical word". G. Cardarelli, *Beyond Oracles – A Critical Look at real-world Blockchains,* Future Internet, 2022, p. 175, also takes up the concept, writing that "[o]racles act as a bridge that can digest external and non-deterministic information into a format that a blockchain can understand"; similarly, B. Curran, *What are Oracles? Smart Contracts, Chainlink & "The Oracle Problem,"* in *medium.com*, 2019.In the ELI Principles, an oracle is defined as a "service that updates a distributed ledger (e.g. a blockchain) using data from outside a distributed ledger system (outside the blockchain context). An oracle transmits information off-chain in computer language to the network".This mechanism may raise critical issues with reference to: (i) the reliability of the information collected by the oracle; (ii) the presence of tools capable of translating technical concepts that are difficult for the program to understand (translation from natural language to programming language); (iv) the jurisdiction in the blockchain context, which is characterized by territoriality in the absence of a specific reference system. For an in-depth analysis of these issues, refer to P. Michaelson, S. Jeskie, *Where the Disputes Lie: When Blockchain Technology Will Need Help Sorting Out Its Contracts, 2021,* available in: https://papers.ssrn.com/sol3/papers.cfm? abstract_id=3893223 *and Z.* Low, *Execution of judgments on the Blockchain: A Practical Legal Commentary, 2021, available at:* https://jolt.law.harvard.edu/digest/execution-of-judgements-on-the-block-chain-a-practical-legal-commentary.

[96] An oracle is any mechanism that extracts data, information or expert knowledge from external sources and provides them to a 'closed' system, i.e. a system that has no access to these sources on its own (inbound oracle); or vice versa, any mechanism that conveys data from a closed system to the external 'world' (outbound oracle). Oracles can be used for transmitting information from the off-chain world to the on-chain one or vice versa (data carried or automated oracles),and for performing computation off-chain and subsequently transmit the outcome on-chain (computation oracles). See, among others: A. Beniiche, A study of blockchain oracles (2020) < https: //arxiv.org/pdf/2004.07140.pdf> accessed 6 Aug 2024; V. Papadouli-V. Papakonstantinou, *A preliminary study on artificial intelligence oracles and smart contracts: a legal approach to the interaction of two novel technological breakthroughs*, Computer Law & Security Review, (2023), 105869.Software oracles can interact with any sources of information available online, such as databases, servers, and websites, and convey data to the blockchain platform in real time.Harware oracles are usually installed in physical objects with electronic sensors, like robots, or relates to objects with QR codes/barcodes; they interact with the physical world and convey the necessary information to the blockchain platform, or vice versa.

[97] See J. ESTCOURT, *Smart Contracts and Dispute Resolution: Faster Horses or a New Car?*, Smart Legal Contracts, (J. Allen and P. Hunn, Eds . Oxford University Press) 2022, pp. 79 – 87,. p. 81: "Because globalisation and digitisation both, at once, enable and require contracts to be negotiated and formed over the Internet, it is difficult to see how contracting parties will not look to online solutions for disputes arising from their agreements. That is to say, that the age has been reached when consumers of dispute resolution will be able to opt for a system based on simplicity, efficiency and economy, and may well be prepared to do so, even at the cost of the loss of intimate involvement in a human managed process and the loss of the imprimatur of a state court". See also: F. Ast – B. Deffains, *When Online Dispute Resolution Meets Blockchain: The Birth of Decentralized Justice,* Stan. J. Blockchain L. & Pol'y, 2021, pp.241 ff, and R. Koulu, *Blockchains and Online Dispute Resolution: Smart Contracts as an Alternative to Enforcement* (2016) 13 SCRIPTed 40.

[98] M. Durovic-A. Jansen *Formation of Smart Contracts under Contract Law,* in Di Matteo, Cannarsa & Poncibo (ed.) Cam- bridge Handbook of Smart Contracts, Blockchain Technology and Digital Platforms (Cambridge U Press, 2020), p. 66; A. Stazi, *Smart Contracts and Comparative Law* (Springer, 2021), p. 81.

[99] Currently, Decentralized Dispute Resolution (DDR) projects are operating on the market that adopt solutions based on human oracles, which make it possible to overcome (in part) the limitations of traditional dispute resolution systems by making use of: (i) the traceability and immutability characteristics of the blockchain; (ii) the self-execution typical of smart contracts; (iii) the flexibility of the intervention of a human agent. However, in these experiments, the efficient use of DDR is limited to medium/low value disputes, which do not require complex technical-legal investigation and which have as their object the mere transfer of sums of money. For a detailed analysis of how some of the current DDRs work, see: Y. Gabuthy, *Blockchain-Based Dispute Resolution: Insights and Challenge,* Games*, 2023,* pp. 14 ff., available *on* https://doi.org/10.3390/g14030034. For a more detailed analysis on the role of blockchain technology with respect to traditional justice, see: P Ortolani, *The Judicialisation of the Blockchain,* available on https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3230880, 2018, pp. 1-39, esp. pp. 30-31, according to whom, "On the one hand, online dispute resolution based on blockchain technologies seems to be a growing reality, meeting a demand of adjudication that cannot be satisfied by state courts due to the excessive costs of litigation. On the other hand, however, traditional avenues of court litigation could become more attractive for users where blockchain technologies become relevant on a wide scale for high-value transactions. In light of this, claims that blockchains are radically disrupting the way justice is administered by courts are excessive".

depending on the circumstances, and also enables dispute resolution through on-chain tools that can be partially automated (e.g., when the oracle is constituted by one or more arbitrators) or even fully automated (e.g., when the oracle is an algorithm applied to one or more databases previously indicated by the parties).

Contracts-on-chain may contain a clause concerning dispute settlement[100] that entrusts an oracle chosen by the parties with tasks that can range from the simple prediction of the outcome of a dispute,[101] to performance of a mediation or conciliation function,[102] to the specification of an arbitration award.[103] In these cases, the parties may also choose the databases that the oracle must access to acquire the information necessary to issue the award. As this additional function becomes more widespread, it becomes more likely that – in the future – smart contracts will, in conjunction with artificial intelligence, automate the oracle function through algorithms, while still adhering to the rules of the European Union that require human control (*human in the loop*) for the use of artificial intelligence.[104] When integrated within a logical platform, the on-chain contract may include a dispute resolution clause that assigns judicial authority (if provided) to the platform or to an entity designated by it, or it may refer disputes to an arbitral tribunal or a judge (off-chain). Similarly, the contract can specify an applicable law that can be applied by the platform itself, by the oracle, by the arbitral tribunal, or by the judge, based on the choice of the parties from various available options.

This feature, offered by the logical platform, addresses the practical issue of establishing a connection between the contract and a legal system, which, as previously mentioned, does not constitute an essential reference for smart legal contracts but becomes central to the contracts-on-chain model. This matter gains significance when considering the current trend in the Web2 environment, where digital platforms have established private legal orders – featuring contracts between the platform and users – that interact with state legal systems and sometimes derogate from their rules.

It is worth looking again at the example of an escrow contract. Formulating this agreement according to the contracts on chain model empowers users to review on the blockchain, at any time, the dispute resolution conditions and methods relevant to execution. For instance, consider the following clauses:

It is understood that the Depositary will refrain from any subsequent payment if the Depositary receives one or more objections or oppositions to the execution of the payment, notarized on-chain, as made by

[100] Reference is made to a classic contractual dispute settlement clause. According to D. Allen - A. Lane - M. Poblet, *The Governance of Blockchain* Dispute *Resolution*, Harv. Negot. L. Rev. 2019, pp. 75, : "Contracting parties face a broader governance choice: what is the best institutional governance mechanism to resolve contractual disputes? The ways to govern a contract—that is, to minimize the costs of contracting and facilitating the exchange—range from courts to arbitration".

[101] Depending on the technical characteristics of the underlying blockchain technology, smart contracts can provide different solutions for dispute settlements. It will be up to the parties to represent the one chosen by mutual agreement in the contract. The use of one or the other solution is left to the parties based on the criterion of convenience, with regard for the characteristics of the contract. The issue has been widely discussed by legal scholars, who propose two different approaches to dispute resolution for smart contracts. According to a first approach, smart contracts, operating within the existing regulatory framework on contracts, can be evaluated by judges or arbitrators by applying existing procedures (including ADR). What is necessary is to import (on-chain) traditional contractual principles. In this sense, see: G. Governatori - G. Idelberger - F. Milosevic - Z. Riveret - R. Sartor - G. Xu, *On legal contracts, imperative and declarative smart contracts, and blockchain systems*, Artificial Intelligence Law 2018, pp. 377-409; M. Sokolov, *Smart Legal Contract as a Future of Contracts Enforcement,* Working paper, 2018, available on SSRN: <https://ssrn.com/abstract=3208292>; A. Holden - A. Malani, *Can Blockchains Solve the Holdup Problem with Contracts?* Working Paper No. 2018-12, University of Chicago. The second approach considers smart contracts as legal instruments distinct from traditional contracts. It follows that the rules existing in various legislation, in both the common law and civil law realms, would not allow for a resolution of disputes that had arisen from the use of smart contracts. In this case, the scholarly proposals go in the direction of promoting a "distributed jurisdiction", a dispute settlement governance based on the blockchain. See: W. Kaal – C. Calcaterra, *Crypto Transaction Dispute Resolution,* Business Lawyer, 2017, pp. 109-153. Some scholars believe that both approaches have limitations and advantages and that a coherent system should be conceived, one capable of containing all possible manners of resolving disputes. See D. Allen - A. Lane - A. Poblet, *The Governance of Blockchain Dispute Resolution,* Harv. Negot. L. Rev., 2019, pp. 75 ff.. According to these authors, blockchain-based dispute resolution systems might not only service the blockchain industry and smart contracts, but also extend into servicing dispute resolution for traditional legal contracts. Contracting parties in a more conventional contract might determine that some blockchain-based form of dispute resolution economizes the costs of dictatorship and disorder. See also B. Howell - P. Potgieter, *Uncertainty and dispute resolution for blockchain and smart contract institutions*, Journal of Institutional Economics, 2021, pp. 545–559, spec. p. 547: "for efficiency-raising objectives to be attained, a role will continue to exist for traditional contract governance institutions (notably contract law and the courts) as complements to blockchain governance arrangements". According to some, however, the use of external courts and arbitrators is the only viable way to terminate contractual agreements that do not take place as originally planned by the parties. Courts and arbitrators have the possibility to apply principles and use evidence relating to subsequent phases with respect to the time of conclusion of the contract, without limiting themselves to the use of processes based on prescriptive rules. See also B. Howell - P. Potgieter, 551–554: "An ex post principles-based court or other adjudication and arbitration process (e.g. alternative dispute resolution processes) taking account of unanticipated changes that occur after a contract is agreed appears to be diametrically opposed to rules-based smart contracting where it is presumed that all possible contingencies can be anticipated ex ante and programmed into code, and be enacted without a change in the future".

[102] It is reasonable to forecast a development of this function by an oracle directly on chain in the insurance sector. The insurance company would insert on-chain settlement proposals which would simultaneously initiate a procedure allowing the parties to resolve the claim by mutual agreement.. If the parties reach an agreement, their choice automatically migrates onto the blockchain and is no longer contestable. Absent any agreement between the parties, and where the maximum number of attempts provided for by the platform has been reached, the system would automatically initiate the litigation procedure.

[103] This is called a "software oracle". Oracles, in fact, can be classified differently on the basis of the source of the data they use: (i) a "software oracle", when the data comes from online sources or when it comes from digital information in general; (ii) a "hardware oracle", when the data transmitted by the oracle originates from the physical world; (iii) "human oracles", when the data entered into the blockchain has previously undergone evaluation or interpretation. A smart contract may include an arbitration clause by which the parties entrust a third-party arbitrator with the dispute settlement. The external information retrieved by the oracle (hardware oracle) would therefore consist of the arbitration award, and the logic of the smart contract would guarantee the execution of what is established therein. In this regard, see Fr. Ortolani, *The impact of blockchain technologies and smart contracts on dispute resolution: arbitration and court litigation at the crossroads,* in Unif. L. Rev., *2019*, pp. 430 – 448, spec. pp. 437 – 442, according to whom, the possibility for parties to encode their contracts into script extends beyond the rather narrow limits originally imposed by the bitcoin protocol, and, thus, new prospects for efficient arbitral procedures arise.

[104] On July 2024 the European Union's Artificial Intelligence Act, Regulation (EU) 2024/1689 ("EU AI Act") was published. According to V. Papadouli-V. Papakonstantinou, (note 96), the interconnection between smart contracts and artificial intelligence takes place through oracles, which can be, among others, highly sophisticated artificial intelligence systems (autonomous systems). The A. also indicate the appropriate legal directions in case of artificial intelligence oracles' failures, based on the most prevalent current approaches to AI's (the user's) contractual and/or non-contractual liability.

the Depositor or the Beneficiary individually or jointly by the Principals. The Deposited Amount will remain deposited in the Deposit Account, which will subsequently be released only and exclusively upon the joint indication of the Depositor and the Beneficiary].

1. GOVERNING LAW AND DISPUTES

10.1. The Agreement is governed by Italian law.

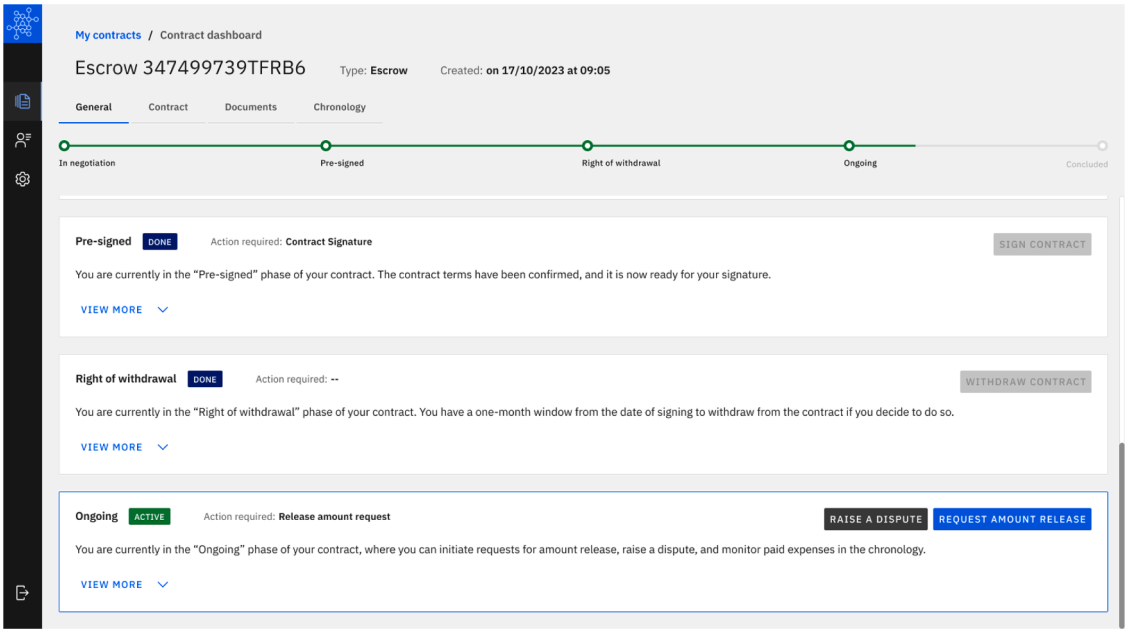
*10.2. Any dispute arising from the Agreement, or from any agreements enclosing, amending and/or supplementing the Agreement, shall be submitted to an arbitration panel composed of three arbitrators (the "**Arbitration Board**"), one of whom shall act as Chairman, appointed by the Parties in accordance with the National Arbitration Rules of the National and International Arbitration Chamber of Milan, which the Parties declare to be aware of and accept in full.*

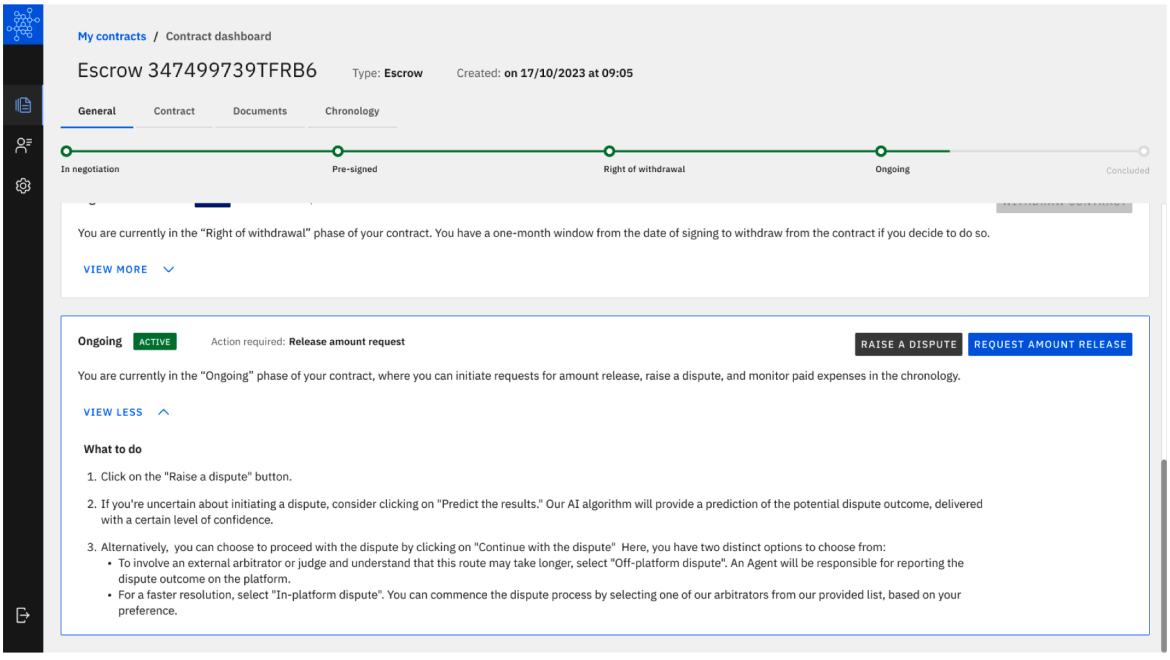
10.3. The arbitrators will proceed in accordance with the law and will decide by applying Italian substantive law; therefore, the decision of the arbitrators shall be a judgment between the Parties. The Parties expressly agree that the arbitration decision cannot be appealed. The Arbitration Board shall also decide on the basis of the loser-pay principle, with regard to the costs of litigation, without prejudice to the right to compensate them.

10.4. The arbitration shall be based in Milan and shall be conducted in the Italian language.

10.5. Without prejudice to the foregoing, it is agreed that any judicial proceedings not falling within the jurisdiction of the Arbitration Board, in any case relating to this Agreement, will fall under the exclusive jurisdiction of the Court of Biella.]

If the contract so provides, the contracts-on-chain platform will offer the user the possibility of selecting the "initiate a dispute" button to refer a dispute to an external off-chain party.





15. Conclusion

The contracts-on-chain model introduces a negotiation process that enhances traditional contracts in several significant regards.

We have verified the operation of contracts-on-chain in a specific environment which we consider ideal for maximizing its potential:

(a) as infrastructure, we utilized a third-generation public blockchain characterized by decentralized consensus, based on pure proof of stake, reduced transaction costs, scalability, high security, and allowance for the modification of smart contracts;
(b) a logic platform serves as the superstructure of the blockchain, the advantages of which significantly mitigate the risks and limitations of public blockchain: it ensures prior identification of parties (whitelisting) and linkage between different logical platforms, enabling what we term 'distributed finance'.

The content of the contract recorded on the blockchain is expressed in natural language, thereby resolving the issue of translation. Machine language, typically used in programming, is confined to self-executing components. The certification of these self-executing elements can be assured by complying with standards defined by regulatory authorities for on-chain contract applications in regulated markets or through third-party certification.

We consider contracts-on-chain as a 'bridge' tool between Web2 and Web3 because it allows for a modular use of the blockchain, ranging from minimum to maximum use, depending on the type of contract and the parties' intentions. Indeed, parties can make minimal use of the blockchain, for example, by utilizing it solely for timestamping – to notarize activities during the pre-contractual, contractual, and post-contractual phases – or they can make maximum use, utilizing all the tools that the blockchain offers to negotiate, conclude, and execute the contract. Between these two extremes, there are various partial uses of the blockchain, useful for notarizing the completion of one or more activities. Due to this modularity in the use of the blockchain, based on the parties' choice, contracts-on-chain significantly reduces the issue of contract inflexibility on the blockchain.

Contracts-on-chain permits the verification of the negotiation phase, *ex post*. The blockchain captures not only the content of the contract but also the entire formation process.

The expression of consent is also securely recorded by the blockchain, thanks to integration with existing digital signature tools, including advanced options, or through new-to-blockchain native identification tools. Furthermore, contracts can be interconnected and executed in a coordinated manner, all within the blockchain environment. Fourthly, the self-executing part of the contract operates according to the automation mechanisms of the public blockchain, functioning entirely independently when events are on-chain and only partially independently when conditions are off-chain. Fifthly, dispute resolution mechanisms can be executed fully or partially on the blockchain, depending on the desired level of autonomy assigned to artificial intelligence. Sixthly, the issue of jurisdiction can also be addressed within the blockchain. This is made possible by the specific environment (i.e. the logical platform), as it is one in which contracts-on-chain operate especially effectively on public blockchains

Thus, it is evident that contracts-on-chain possess distinct characteristics when compared to smart legal contracts. It can also be observed that the new issues raised by contracts-on-chain and the way they resolve old ones are topics warranting further exploration by academics.

It is apparent, therefore, that contracts-on-chain can substantially address many of the questions that legal scholars have raised in recent years regarding the nature and function of smart legal contracts. Unlike the smart contracts and smart legal contracts previously scrutinized by scholars, contracts-on-chain are not mere execution mechanisms for contracts established elsewhere, nor do they simply represent the phase of contract stipulation on the blockchain. By means of the contracts-on-chain process, contracts are enhanced by the blockchain and are effectively transformed into '*supercontracts*'.

The latest generation of public blockchain enables the definitive identification of parties in a certain and permanent manner; it records and preserves negotiations; it features the automatic execution of contract clauses (including payments) upon predefined conditions; it interlinks and immutably records multiple contracts; it records parties' certified signatures using established methods or new ones native to blockchains; and it employs a dispute resolution mechanism based on the parties' agreed-upon legal choices.

A logical platform integrated into the blockchain aligns the ecosystems in which parties operate especially closely with the regulatory model that is still adopted by European supervisory authorities particularly in the banking, insurance, and financial sectors. This ensures an adequate level of protection without compromising certain aspects of enhanced efficiency, security, and decentralization that are inherent in public blockchain.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

Fabio Bassan reports administrative support was provided by Roma Tre University. Fabio Bassan reports a relationship with Roma Tre University that includes: employment. Fabio Bassan has patent pending to No NO If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The authors do not have permission to share data.