EUBlockchain
Observatory and Forum

Smart Contracts

About this report

This is the eleventh of a series of reports that will be published addressing selected topics in accordance with European Commission priorities. The aim is to reflect on the latest trends and developments and discuss the future of blockchain in Europe and globally.

This report has been produced by the EU Blockchain Observatory and Forum Experts Panel and team.

The EU Blockchain Observatory and Forum team:

- Alexi Anania, Ash Costello, Agata Ferreira, Sergio Gonzalez-Miranda, Amit Joshi, Jim Mason, Lisa Trujillo – EU Blockchain Observatory and Forum Expert Panel
- Marianna Charalambous – IFF, University of Nicosia
- Konstantinos Votis, Kristina Livitckaia, Iordanis Papoutsoglou – CERTH
- Nikolaos Kostopoulos, Tonia Damvakeraki – Netcompany Intrasoft

Special thanks to ***Dr Ioannis Revolidis****, University of Malta for his insightful contribution and overall support in the authoring process.*

Special thanks to **Scope** for the editorial review and language proofing.

Note

While we have done our best to incorporate the comments and suggestions of our contributors where appropriate and feasible, all mistakes and omissions are the sole responsibility of the authors of this report.

Disclaimer

The information and views set out in this publication are those of the author(s) and do not necessarily reflect the official opinion of the European Commission. The Commission does not guarantee the accuracy of the data included in this study. Neither the Commission nor any person acting on the Commission’s behalf may be held responsible for any use which may be made of the information contained herein.

Smart Contracts

Table of Contents

ABOUT THIS REPORT ... 2

CHAPTER 1: DEFINING SMART CONTRACTS ... 5

1.1 What is a Smart Contract? ... 5
1.2 How Do Smart Contracts Work ... 6
1.3 Why Use Smart Contracts: Key Benefits and Limitations ... 7
1.4 How to Safeguard Against Security Vulnerabilities? ... 10
1.5 Cost of Implementation and Errors ... 11
1.6 Legal Ramifications & Adaptability ... 11
1.7 Debugging Smart Contracts ... 11
1.8 Consumers ... 13

CHAPTER 2: STATE OF PLAY & USE CASES ... 14

2.1 State of Adoption ... 14
2.2 Legal Considerations and Paradigm from Regulators Across Europe ... 20
2.2.1 The Data Act *20*
Smart Contracts as defined in the Data Act ... 20
Requirements for Smart Contracts under the Data Act ... 20
2.2.2 National approaches *21*
Italian Law on Smart Contracts ... 21
UK Law on Smart Contracts ... 21
2.3 Thematic Use Cases ... 23
2.3.1 DAOs *23*
2.3.2 NFTs *24*
2.3.3 Identity Credentials & Data Management *25*
2.3.4 Decentralized Finance *26*
2.3.5 Synthetic Tokens & Indexes *26*
2.3.6 Renewable Energy tokens *27*
2.3.7 Enforcing/Billing Usage Agreements *27*
2.3.8 Emerging Technologies *28*

CHAPTER 3: HIGHLIGHTS AND CONCLUSIONS ... 29

REFERENCES ... 31

WEB SOURCES: ... 32

ENDNOTES ... 33

Table of Figures

Figure 1: Comparing Smart to Traditional Contracts 8

Figure 2:New Verified Smart Contracts Period 01/01/202- 20/10/2022, Source: EUBOF 14

Figure 3: Contracts uploaded on Ethereum blockchain with verified source codes only, Source: Etherscan.. 15

Figure 4: Contracts uploaded on Binance Smart Chain blockchain with verified source codes only, Source: BSCScan 16

Figure 5: Contracts uploaded on Polygon blockchain with verified source codes only, Source: PolygonScan 16

Figure 6: Contracts uploaded on Tron blockchain, Source: TronScan........ 17

Figure 7: Contracts uploaded on Solana blockchain, Source: SolScan 17

Figure 8: Contracts uploaded on NEAR blockchain, Source: NEAR Explorer 18

Figure 9: Contracts uploaded on Aurora blockchain with verified source codes only, Source: Aurora Scan... 18

Figure 10:Contracts uploaded on Optimism blockchain, Source: Optimistic 18

Chapter 1: Defining Smart Contracts

1.1 What is a Smart Contract?

A smart contract is an event-driven programme that runs on a decentralized, distributed shared and replicated ledger, which can take custody over and instruct the transfer of assets on that ledger.

The concept of ‘smart contracts’ is not entirely new. It was first given its name in the 1990s by computer scientist and legal theorist Nick Szabo. In 1998, Nick Szabo defined smart contracts as computerised transaction protocols that execute terms of a contract[1]: “A Set of promises specified in digital form, including protocols within which the parties perform on these promises”. In the past, Nick Zsabo has given the example of a vending machine as a crude attempt at defining a smart contract[2] i.e., the vending machine ‘understands’ what the deterministic outcome should be once a specific set of inputs is fulfilled. The mechanism of a vending machine is based on if-then logic, in which payment triggers irrevocable actions where money is retained and an item is supplied. Once this set of actions is triggered by the insertion of the coin, it cannot be stopped or reversed. Performance occurs automatically, without human intervention. This sequence of events is pre-programmed and embedded into the code of a vending machine.

One more point to notice in the parallelisation of smart contracts with vending machines is that access to the services does not require any validation of the user’s identity. Essentially, whoever has the money should be able to treat themselves from the vending machine. The process is highly automated with no authorisation, as the vending machine requires only the user’s monetary funds to execute the process. Similarly, smart contracts can be considered “black-box”, executing a logical process where a given input will have an expected output.

Smart contracts pre-exist blockchain and can exist without blockchain technology, like in the vending machine example. Due to technological limitations, smart contracts have been out of the spotlight for some time. The emergence of blockchain technology brought them back from obscurity and into the mainstream of technological advancement. Blockchain has enabled the progress of smart contracts from simple automated contracts to fully autonomous self-executing and self-enforced contracts built on decentralized platforms and supported by a blockchain ecosystem. Smart contracts combine a number of technological advancements, including electronic contracting, cryptography and tamper-proof and algorithmic executions based on consensus.

In 2014, Vitalik Buterin (Co-founder of Ethereum) explained at a blockchain summit how smart contracts work:

> “Contracts are translated into computer language and stored in blocks. The parties to the contracts, which are copied to distributed ledgers, are kept 100 percent anonymous. The code snippet is ready with specific tasks and details (time limit, what goes where, from where to where, etc.). When the time comes, it takes action to fulfil the transaction, and if the necessary conditions are met, the transaction is successfully completed or cancelled before completion.”[3]

The inception of blockchain coming from Bitcoin’s whitepaper was novel as it suggested a decentralised peer-to-peer token exchange system. This is considered the first generation of blockchain, where the network curates a digital currency for the users. The trustless and decentralised environment can expand to execute the business logic securely. Smart contracts suggested by Ethereum in 2014 triggered the second generation of blockchain. Essentially, a Turing-complete language can allow the creation of arbitrary programs that will be executed by a machine. (EVM in the Ethereum’s case). Similarly to the vending machine, the initial applications focused on building financial solutions, with notable examples coming from the DeFi sector. In the future, more emphasis will be placed on delivering properly functioning decentralised application (DApp). .

There are currently various use cases for the prospect of using smart contracts, some of which display the potential to displace conventional intermediated processes:

- Securities: smart contracts have the ability to disintermediate and reduce counterparty risk as well as operational risk.
- Trade finance: smart contracts can make the process more streamlined, with faster execution and delivery times.
- Financial data recording: smart contracts can increase accuracy and transparency.
- In Rhode Island, complex business licensing processes were simplified, faster and automated using blockchain and smart contracts for digital identity and credentials

…and many more.

Although it is now largely accepted that not all smart contracts have legal relevance, some members of the technical community oppose the use of this terminology due to unnecessary legal/regulatory scrutiny and potential confusion as to their legal significance. Even Vitalik himself, said he regretted adopting the term ‘smart contracts’ rather than more technical terms, like persistent scripts for example. It has been suggested that if the term used from the start had not included the legally significant word ‘contract’, but had instead been called, for example, ‘smart code’ or ‘computerised protocol’, it would be clearer that in general, smart contracts are not contracts in the legal sense, although nothing prevents them from having legal effects (Ferreira, A).

1.2 How Do Smart Contracts Work

From an agnostic point of view regarding its underlying technology, smart contracts are pieces that contain computer code that are entrusted with completing a task.

The code contains the rules that define the conditions under which the smart contract has to act and how it has to behave. Their domain of action is internal to the blockchain that contains them. Although they can also receive information from external sources through the use of oracles. This task is typically known as a transaction which must be carried out as long as the conditions specified within the computer code are met (satisfied agreement).

The execution of the transaction does not involve the intervention of any intermediary. Participants must cryptographically sign their participation.

- Smart contracts have input interfaces that receive input data.
- The data is interpreted and validated according to the programmed rules.
- The signatures of the parties involved in the transaction are verified.

Within the set of smart contracts of a blockchain, a smart contract can act in isolation from the others or interact with them to complete the tasks entrusted to it. In this way, suites of smart contracts for specific purposes can be designed. Thus, a Smart contract can focus on receiving data from the outside, while another smart contract can focus on processing business rules. Finally another smart contract can take care of giving persistence to the result (the smart contracts process their instructions and eventually produce a result to be stored within the blockchain).

The operation of a smart contract is similar to other **blockchain** transfers. These are the necessary steps:

Step 1: A user initiates a transaction from their **blockchain** wallet;

Step 2: The transaction arrives at the distributed **database,** where the identity of the user wallet is confirmed;

Step 3: The transaction, which may be a **transfer** of funds, is approved; the transaction includes the **code** that defines what type of transaction is to be executed;

Step 4: The transactions are added as a **block** within the blockchain;

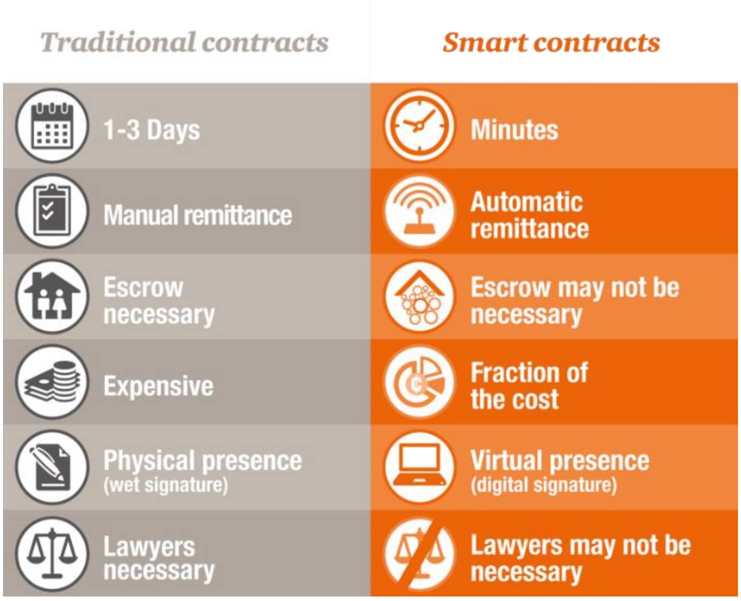
Step 5: Any change in **contract** status follows the same process to be updated.

Some of the most widely used platforms to develop and execute smart contracts on blockchain include:

- **Ethereum:** smart contracts are written in a programming language called Solidity and are being executed by the Ethereum virtual machine. Currently, Ethereum is considered the most popular platform. The Ethereum Virtual Machine is a computation engine responsible for deploying and executing smart contracts and computing the state for every new block added to the Ethereum blockchain. There are numerous of Ethereum Virtual Machine compatible blockchains executing smart contracts that enable developers to easily transfer their smart contracts across different EVM compatible networks. There is a long list of projects that have launched an EVM-compatible blockchain. Among the most popular EVM Compatible blockchains are Avalanche, Binance Smart Chain, Arbitrum, Polygon, Optimism, Harmony, Fantom and Celo Network. An extended list of all EVM compatible blockchains is provided by DefiLlama.
- **Hyperledger:** is an open source system developed by the Linux Foundation and is not a blockchain, but a flexible platform on which smart contracts can be developed.
- **Polkadot:** it is an alternative to blockchain and is famous for its ability to host parachains, i.e., chains within chains, allowing more transactions than usual.
- **Solana:** Solana smart contracts can be created using multiple programming languages. While the native Solana smart contact language is Rust, the protocol also supports smart contract development in C++ and Solidity, along with support for other languages through third-party JSON RPC API SDK clients.
- **Cosmos:** Cosmos is an expanding ecosystem of independent interconnected blockchains connected through the Inter-Blockchain Communication protocol. Developers can choose to build autonomous application-specific blockchains that can easily interconnect. The standard protocol for inter blockchain communication, IBC allows blockchains in the ecosystem to connect so that they can transfer tokens and other data between each other frictionlessly and seamlessly. Cosmos currently has three different SDKs that allow developers to write smart contracts in Javascript, Rust, or Solidity.
- **Stellar:** Stellar was founded in 2014, making it one of the oldest smart contract platforms. It is maintained by the Stellar Development Foundation and has been repeatedly proclaimed as one of the most exciting blockchain startups out there.

1.3 Why Use Smart Contracts: Key Benefits and Limitations

Smart contracts are much more than digital cousins of paper based contracts and we are just about scratching the surface with real world implementations.



Traditional contracts	Smart contracts
1-3 Days	Minutes
Manual remittance	Automatic remittance
Escrow necessary	Escrow may not be necessary
Expensive	Fraction of the cost
Physical presence (wet signature)	Virtual presence (digital signature)
Lawyers necessary	Lawyers may not be necessary

Figure 1: Comparing Smart to Traditional Contracts

Some of the most noteworthy advantages of smart contracts are as follows:

Speed & Efficiency

With a high degree of automation and self-execution of the terms agreed upon, smart contracts streamline the entire contract implementation lifecycle thereby helping in fast, seamless and efficient execution. Some of the easily attainable advantages smart contracts help enable are automated payments, insurance claims, transparent and efficient supply chains, efficient corporate governance, or streamlining of data management for clinical trials. Efficiency gains from smart contracts include cost reduction, time savings, and overall operational improvements.

Transparency

Smart contracts help ensure higher transparency and at the same time lower chances of corruption since any changes made to the contract require the consent of all parties involved or at the least can be traced back easily. The scope of manipulation or non-performance due to any individual is highly unlikely.

Removal of Intermediaries & Cost Effectiveness

By removing the need for third parties or intermediaries for execution, smart contracts eliminate the risk of manipulation. Cost savings are derived from the removal of intermediary layers in relationships and peer-to-peer transacting. Automatic performance of smart contracts potentially eliminates the need for institutional enforcement and presents a cheaper and more effective alternative for ex-ante guarantee of performance. Using smart contracts results in the elimination of errors that occur due to manual completion of numerous forms.

Security

Due to the use of data encryption, smart contracts are tamper-proof and a highly secure alternative to paper-based contracts. Smart contracts are pieces of code which can be readily reused for similar operations with minimal changes as per requirements. It provides the advantage of using an already tested piece of code to build upon. Some of the platforms working on smart contract security are listed below:

Open Zeppelin provides security products to build, automate, and operate decentralised applications with the Openzeppelin Helper Library which is built on a solid foundation of community-vetted code.

i. Implementations of standards like ERC20 and ERC721.

ii. Flexible role-based permissioning scheme.

iii. Reusable Solidity components to build custom contracts and complex decentralised systems.

Why3 is a platform for deductive programme verification. It provides a rich language for specification and programming, called WhyML, and relies on external theorem provers, both automated and interactive, to discharge verification conditions

Oyente, a smart contract auto-auditing tool, is used to analyse smart contracts and returns possible bug within it

On the other hand, there are different limitations for smart contracts, which depend on the perspective that the aspect being researched. For example, the Ethereum blog provides two technical limitations existing in native smart contracts. The first one concerns the **inability of smart contracts to evaluate real-world events**. This fact should not come as a surprise since blockchains are generally independent and separate environments. The separation is a mechanism to guarantee the network's security on the basis of the consensus algorithm. As a result, data residing in databases ranging from employee information to weather data or football game results are data outside of the blockchain network that are considered potentially dangerous for the network. There are ways to mitigate this limitation, as oracles are a solution for bridging trusted data to the blockchain. In particular, Chainlink's blog points out blockchains' isolation and the potential for hybrid smart contracts to bridge this weakness by using oracles. The second limitation from the Ethereum blog references **the maximum contract size posed on smart contracts**. The limitation is defined by protocol EIP-170 which prevents denial-of-service attacks by executing never-ending contracts to congest the network. Developers are bound to include functionalities that do not exceed the maximum of 24 KB, and the mitigation proposed is the Diamond Pattern.

We are in the early stages of smart contract implementations and, as a result, we find a few limitations which hinder with a higher adoption rate. Some of them are listed below.

Smart Contract Vulnerabilities & Security Lapses

The majority of smart contracts on the first generation of blockchain platforms are balance-based, whereby a defect in one exposes all addresses linked with that contract. As a result of if hackers discover a bug and are successful in exploiting it, they will be able to drain cash from every single address that has ever transacted with the faulty system.

This is where security and smart contract auditing companies come to the fore to mitigate the risks of known issues and aid projects - either building new smart contracts or using existing ones - to avoid any known bugs or vulnerabilities.

Decentralised finance (DeFi) is a highly volatile market where millions of people become victims of large-scale privacy breaches and theft. If we look at the number, we find that within the first three months of 2022 USD 682 million was lost due to hacks and crypto fund owners and businesses lost USD 3.3+ billion as a result of hacker attacks and security breaches. Deliberate hacks that exploit blockchain-type technologies can include ‘rug-pulls’, ‘flashloan attacks’, and a combination of these attacks with traditional types of irregular behaviour. Opportunistic attacks on smart-contracts can also exploit vulnerabilities, bugs and errors in the contract code.

Some of the smart contract related breaches are mentioned below:

> The tPoly Network attack in February 2022 involved the compromising of smart contracts in three blockchains: BSC, Polygon, and Ethereum resulting in a combined loss of USD 611 million. The hacker was able to siphon the funds by exposing the security flaws in Poly's unverified contracts to demonstrate the magnitude of risks they created, but later returned the funds.
>
> Grim Finance, a yield optimizer protocol, suffered a re-entrancy attack costing itUSD 30 million in December 2021. As a result of a re-entrancy attack, the hacker managed to feed a series of fake additional deposits in the system while the previous ones were still incomplete. This allowed the hacker to release Fantom tokens for USD 30 million, thus exposing the absence of a re-entrancy guard on the platform. The smart contract audit firm Solidity Finance erroneously identified the guard as active.
>
> Hackers repeatedly used the flash loan vulnerability of Cream Finance's architecture to drain funds from the system. A flash loan hack is an attack involving the receipt of a non-collateralized loan due to token pair price manipulations. The August attack was the largest in scale, resulting in a cumulative loss of USD 180 million.
>
> A "rug-pull" is a form of fraud where scammers entice investors into a seemingly attractive network, and then remove the funds or remove the ability for investors to exit. This means that investors' funds have been misappropriated, or investors cannot access their funds. The scammers take the inaccessible funds. Meerkat Finance is an example of a 'rug-pull' attack. The first day after the project's launch, USD 31 million was lost to scammers. It transpired that the code was updated just before the attack to give the developers backdoor access to investors' funds. The next day (i.e., the second day after the project's launch) the investors were refunded. Apparently the entire project was a stunt to illustrate how easy these hacks are to stage. Alchemix suffered another form of rug-pull attack when the contracts for one of their synthetic assets allowed users to withdraw their collateral without affecting their loans. This resulted in a 'rug-pull by community' costing USD 6.5 million.

1.4 How to Safeguard Against Security Vulnerabilities?

Security Audits

The importance of an audit before the launch of a smart contract has been made evident by the multiple hacks and exploits that have occurred in the last few years. Audits help test your smart contracts for immunity to various attacks, due to issues like timestamp dependence, weak protocol code, and malicious external calls. It is also essential to get audits conducted after updates to test smart contracts for all errors and issue a report with identified vulnerabilities and improvement recommendations.

Penetration Tests

By organising penetration tests, one can test your smart contracts' immutability and hack immunity. The pen test can cover APIs, front and back-end servers, or smart contracts. It is a form of ethical hacking; in other words, a security audit firm organises a controlled attack on the smart contracts to see whether it stands or cracks. It can enhance firewalls to anticipate real-life attacks based on the outcome of the attack.

Testing Frameworks

- Foundry : In Foundry, tests and scripts are written in Solidity.
- Hardhat: In Hardhat, tests and scripts are written in Javascript.

- Brownie: In Brownie, tests and scripts are written in Python.

The most common tools used for testing security vulnerabilities include:

- Slither detects vulnerable Solidity code with low false positives.
- Echidna is a programme designed for fuzzing/property-based testing of Ethereum smart contracts.
- Eth Security Toolbox is a Docker container preinstalled and preconfigured with all of Trail of Bits' Ethereum security tools

1.5 Cost of Implementation and Errors

There are costs associated with implementing smart contracts, for example, the potential costs of switching to a smart contracts network and persuading counterparties to participate in that network. Minimisation of human intervention and formalisation of smart contract creation results in not only cost savings but also in collateral costs related to coding errors, the implications of immutability, and the need to reverse unintended transactions.

While Altay & Motawa researched the applicability of smart contracts in the construction sector, they provided a list of the limitations of smart contracts. The list includes the difficulty in changing transactions, updating relations for a long-term trade, restrictive execution options, and legal responsibility, but the intriguing subject is hacking and fund security. There are different security issues to consider while deploying smart contracts. Systems can be vulnerable to hacks due to oversights during development. Bugs can go unnoticed, and the deployed services may be susceptible to malicious hackers. For this reason, philosophies like DevSecOps and Continuous Integration are developed along with tools to automate the tests and reports. There are tools for debugging smart contracts, which are to be presented in a subsequent subchapter. Moreover, it should be noted that smart contracts are addresses identical to users' digital wallets and correspondingly manage digital assets.

1.6 Legal Ramifications & Adaptability

Since smart contracts reduce dependence on intermediaries and lawyers, it becomes imperative for all involved parties to be aware of the legal ramifications of compliance and regulatory norms. But the language used in smart contracts is not the same as in legal scenario. Multiple challenges may arise to clarify the governing law maker's perspectives and the jurisdictional stance. Additionally, comprehending the adequacy of protection of personal data stored on the blockchain is an important question which requires stringent KYC/AML gateway compliance.

Changing smart contract processes is almost impossible, any error in the code can be time-consuming and expensive to correct. From the legal standpoint, logic-based execution might be a painstaking process since contract construction is subjective, and phrases like "good faith" and "best efforts" are intentionally included to leave room for flexibility. Going forward, smart contracts have to become flexible to accommodate adequately the contractual safeguards incorporated in an appropriately developed body of laws to address real world issues such as tech failures, attacks, war/natural disasters, crime, tax, and changing legislation to enhance adaptability.

1.7 Debugging Smart Contracts

Software releases can hit production while bugs that can be exploited are in the code. Software bugs can be errors or oversights, meaning they are not necessarily the result of malicious actions. The severity of bugs for the system can vary as they can go unnoticed for long periods with no consequences for the system's operations. An example comes from Consensys' Github where a range of tools detecting vulnerabilities for Ethereum smart contracts perform different tests. The tools are categorised into visualisation, static and dynamic analysis, classification, testing, and linters.

There are reported cases of bug exploitations in the blockchain that have cost significant amounts for the protocols. Exploitation recent to the current report was reported in Transit Swap, a DEX aggregator, costing USD 21 million. The most documented case of an exploit due to code vulnerabilities is the DAO Hack. The DAO was launched in 2016 as one of the early decentralised autonomous organisations intending to act as an investment platform for the blockchain sector. The DAO launched on Ethereum successfully raised USD 150 million, but was hacked in the first three months of operation for USD 60 million. The vulnerability was located in the wallet's smart contracts, where a backdoor for draining the wallets existed. A consequence of the DAO hack was Ethereum's hard fork to a state of the blockchain prior to the hack.

As a remedy to bugs and exploits in the code, protocols and organisations take actions to safeguard their operations. One such action is the bug bounty programme, where developers can earn recognition and compensation for uncovering bugs. This is a tactic implemented in software development, and blockchain has adopted it, for example Hedera's programme. There are incidents of payments to white hat hackers reporting vulnerabilities in protocols like the USD 2 million of Aurora. Generally, bug bounty platforms in the blockchain space increase users' security as blockchain adoption matures and more people invest their funds in protocols. Immunefi is a platform dedicated to blockchain and Web3 bugs providing funds to developers for reporting bugs.

As technologies and languages mature with the passage of time, tools are deployed to automate tasks for developers and mitigate risks. These tools are invaluable for the technology, as they save time and allow developers to focus on solving fundamental issues. Security is an issue calling for particular efforts, and the testing automation can help to establish a clear framework for deploying smart contracts. Furthermore, security tends to be a part of development in the earlier stages so that bugs and security issues can be revealed early and mitigation actions taken.

Frameworks are suggested in the literature for automating the processes for pinpointing bugs. Formal verification is a procedure for computer programmes to ensure the satisfaction of certain formal statements. One framework for formal verification of smart contracts on Solidity is the use of Why3 language. Nehai and Bobot set out four phases for testing smart contracts, which are:

- encode smart contracts in Why3,
- formulate specifications and functional properties,
- verify the programme's behaviour on the Why3 system,
- compile the Why3 contracts to Ethereum virtual machine.

Another framework from the literature on security vulnerabilities is proposed by Luu et al. in an attempt to ameliorate security risks by providing intelligence to smart contracts. The tool developed is called Oyente, and the authors explain that this tool and similar ones are necessary because correct reasoning is empowerment and irrelevant to the contract's popularity or invested funds. When presenting a taxonomy of bugs, symbolic execution is the basis for the tool deployed for Ethereum's contracts to address these bugs with testing. Symbolic execution creates the paths created by the smart contract's logic and defines feasible and infeasible paths to determine bugs.

Privacy and multi-party computation are matters that must be addressed in the execution of smart contracts. For this reason, tools like Raziel (Sánchez) implement a variety of methods to ensure their private, correct, and verifiable execution. The aforementioned tool uses Zero-Knowledge Proofs to confirm validity prior to the execution of the smart contract to any third party and keep the information private. Related work is from accountable algorithms (Kroll) where a commit and prove scheme is implemented for zk-SNARKs.

Work is being done on securing smart contracts and eliminating bugs in the code. This section presented some market initiatives and literature-based solutions for testing smart contracts before deployment. The tools developed for testing can help in the maturation of blockchain as testing for vulnerabilities can be automated. Finally, harm to users from bugs can be mitigated if applications market have been tested.

To address these vulnerabilities, the European Commission is proposing Essential Requirements to smart contracts in the Data Act legislative proposal; an overview is given in this documents chapter.

For more details please refer to the Article 30 of the proposal for a regulation of the European Parliament and of the Council on Data Act.

1.8 Consumers

There is a view that smart contracts might not be appropriate consumer contracts unless there are specific measures implemented that allow consumers to understand coded smart contracts and their implications, for example, natural language translation into a form easily readable and understandable by the average consumer. In many jurisdictions, contractual requirements applicable to consumer contracts are subject to specific mandatory laws and regulations, which are usually stricter than in business-to-business contractual dealings.

Consumers are usually granted informational rights, cooling-off periods, and rights to revoke the contract within a specific time frame. Implementing such requirements on an immutable, automated blockchain network could be challenging. Automated contracting using smart contracts has other implications for consumers. Automation might reverse the burden of proof to the consumer’s disadvantage. Retailers would simply receive the payment automatically through smart contract execution, and consumers would have the burden of proof if they claimed that such payment should not have been made.

The party benefiting from such change would be the retailer, and the risk would be shifted onto the consumer, which might not be appropriate or possible. In the current era of new laws and regulations being introduced to protect consumers and individuals against increasingly complex digital goods and services,

it seems unlikely that blockchain solutions that pass burden of proof onto consumers would be acceptable. Even jurisdictions that are more progressive in relation to smart contracts legislation recognise the potential vulnerability of consumers when using blockchain technology. For example, Illinois implemented new blockchain laws that have exceptions to disallow blockchain records for consumer credit defaults, utility cut-offs, health insurance coverage changes, or recall of a product.

Current European Union legislative initiatives mentioned later in this report reflect concern over the weaker party to smart contracts and the need to ensure mechanisms that could stop the execution of smart contracts to protect such vulnerable parties.

Chapter 2: State of play & Use Cases

2.1 State of Adoption

Promising use cases, along with the advances in the technology, have contributed to the significant rise in the adoption of smart contracts. As the blockchain industry has matured, we have noticed that the challenges that have held back the adoption of smart contracts – such as the learning curve, user expectations, and the scalability of blockchain networks – are gradually being resolved. Notable multinational corporations have started using smart contracts for commercial purposes, while in previous years there were only announcements about internal piloting and prototyping projects.

In order to describe the real state of adoption of smart contracts, it is not enough to provide a comparison of existing smart contracts in a blockchain network, since a smart contract could be uploaded but not have a real use case or interaction with real users. It is vital to synthesize a variety of data – such as the state of adoption by the developer community, the total locked value of a protocol, the daily active users, and numerous other factors – to draw conclusions about the real state of adoption. Below, we present some high-level data insights that might give us an outline of the growth in smart contract adoption, while the following report will focus on use-cases and refer to notable success stories.

Verified Contracts

An interesting approach to understanding the growth of smart contracts is to analyse the daily growth of verified smart contracts, such as those provided by trusted block explorers. Below, we compare new verified smart contracts across the Ethereum, Polygon, REUM, Binance Smart Chain, Avalanche, and Optimism Blockchains. We have selected the above networks because the methodology used by their respective block explorers to extract the data is the same, while other blockchain explorers use different methodologies. The verified data about smart contracts should not be confused with the total number of smart contracts deployed on the blockchains. This number is not sufficient to describe the current state of adoption on a particular blockchain but could indicate some patterns that we will aim to analyse.

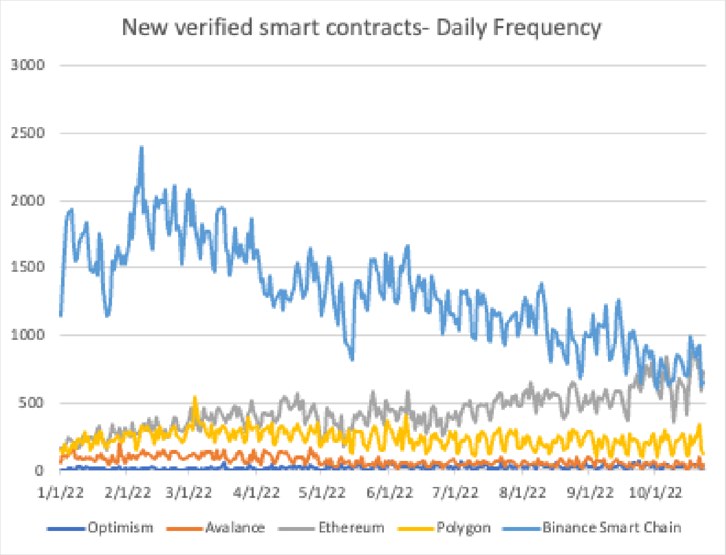


Figure 2:New Verified Smart Contracts Period 01/01/202- 20/10/2022, Source: EUBOF

The total number of smart contracts deployed on the Ethereum blockchain is over 4,658 million as of 24 October 2022, with datasets extracted by Dune Analytics.

Below we can view the daily verified contracts on the Ethereum network. The new daily verified contracts during 2022 are between 400-850 during 2022.During 2021 this figure did not exceed the 400 milestone.

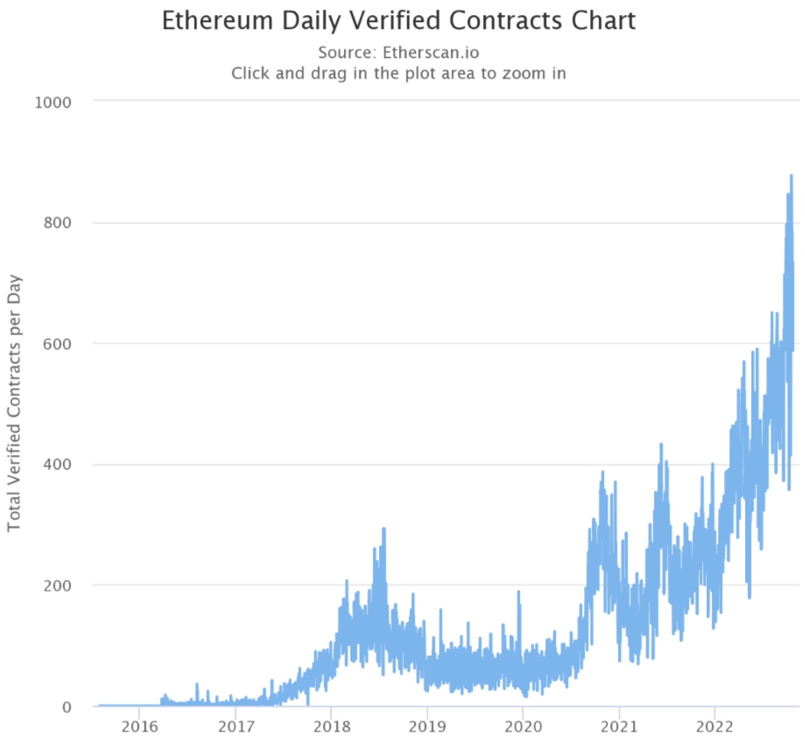


*Figure 3: Contracts uploaded on Ethereum blockchain with verified source codes only, Source: **Etherscan***

Binance Smart Chain is showing a significantly higher number of daily verified contracts, though in comparison to Ethereum it is showing declining adoptions during 2022, with the new creation of contracts having taken place during 2021, potentially attributable to the skyrocketing prices of cryptocurrencies in the given period. The new contracts verified on Binance Smart Chain are almost double in comparison to the Ethereum network, which could be explained by the significantly lower transaction fees on the blockchain.

.

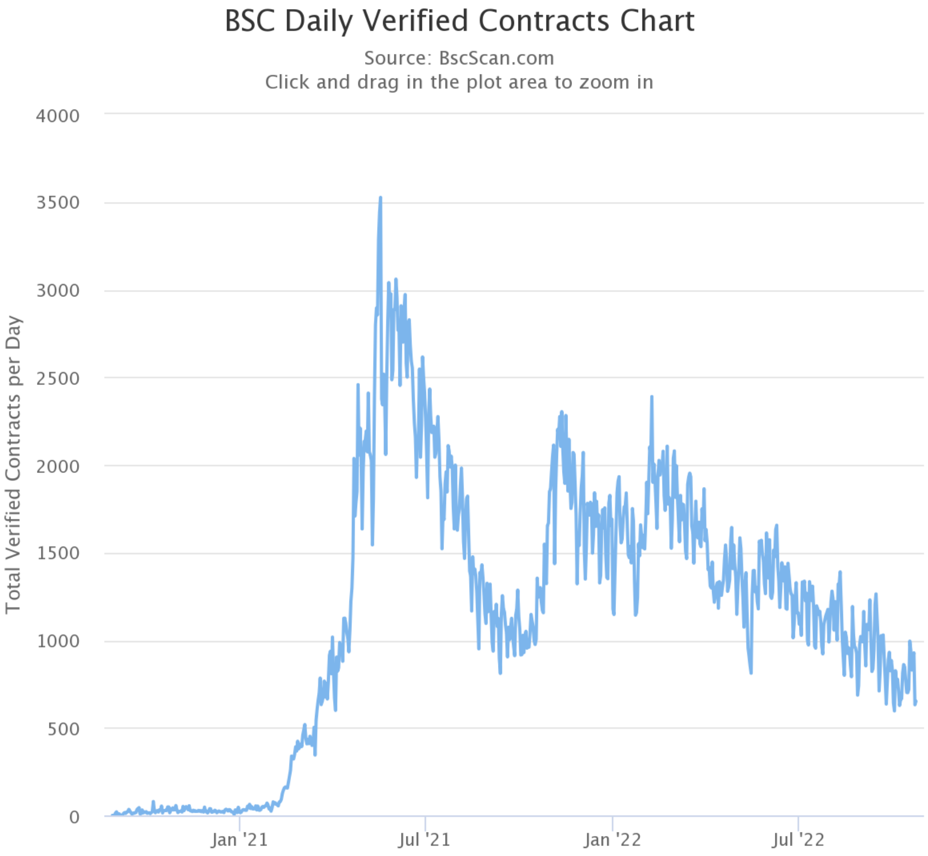


Figure 4: Contracts uploaded on Binance Smart Chain blockchain with verified source codes only, Source: [BSCScan]

The new verified smart contracts on the Polygon blockchain follows a more linear progression, varying between 100-150 on a daily basis, with only July 2021 constituting abnormal activity on the network which appears to be bot-enabled smart contract deployment.

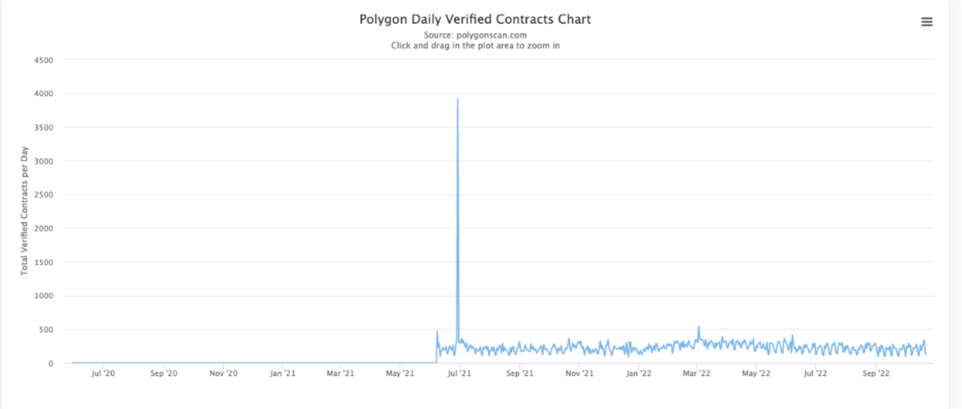
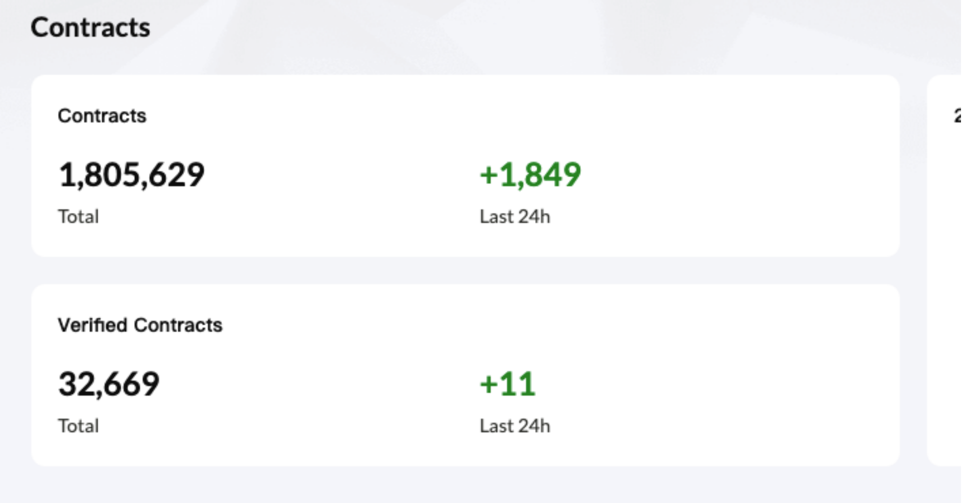


Figure 5: Contracts uploaded on Polygon blockchain with verified source codes only, Source: [PolygonScan]

Other blockchain explorers are providing the total number of smart contracts ever deployed on the network with great accuracy. As a reference, TronScan is reporting a total of 1,805,629 smart contracts uploaded on the Tron blockchain, with a proportion of 32.669 being verified contracts. This number provides us with the metric that only 0.02% of the total deployed smart contracts on Tron are verified. While verified does not mean “trusted” contracts, we can argue that verified smart contracts are more reputable and more likely to reflect better the state of adoption in a blockchain network.



Contracts

Contracts

1,805,629
Total

+1,849
Last 24h

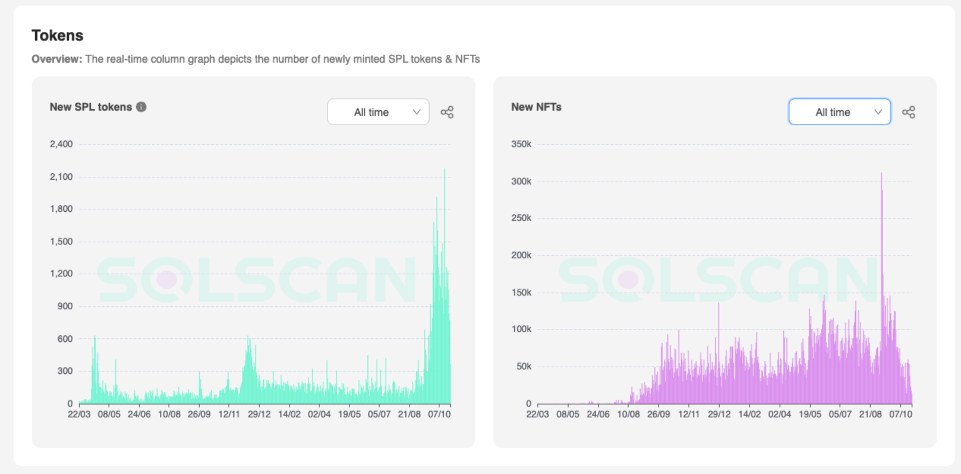
Verified Contracts

32,669
Total

+11
Last 24h

*Figure 6: Contracts uploaded on Tron blockchain, Source: **TronScan***

SolScan provides 2 different metrics that help us understand the total smart contracts on the Solana blockchain. When it comes to the Solana blockchain, we can immediately identify a skyrocketing activity related to NFT smart contracts.



Tokens

Overview: The real-time column graph depicts the number of newly minted SPL tokens & NFTs

*Figure 7: Contracts uploaded on Solana blockchain, Source: **SolScan***

A valuable metric is provided by the NEAR Explorer, which reports the total number of unique smart contracts deployed on the network. The number of unique contracts deployed on the NEAR blockchain total 5,523. For the same period, the total number of contracts deployed on the network is 55,500 which provides with the very useful insight that approximately 10% of the smart contracts deployed on NEAR are unique.

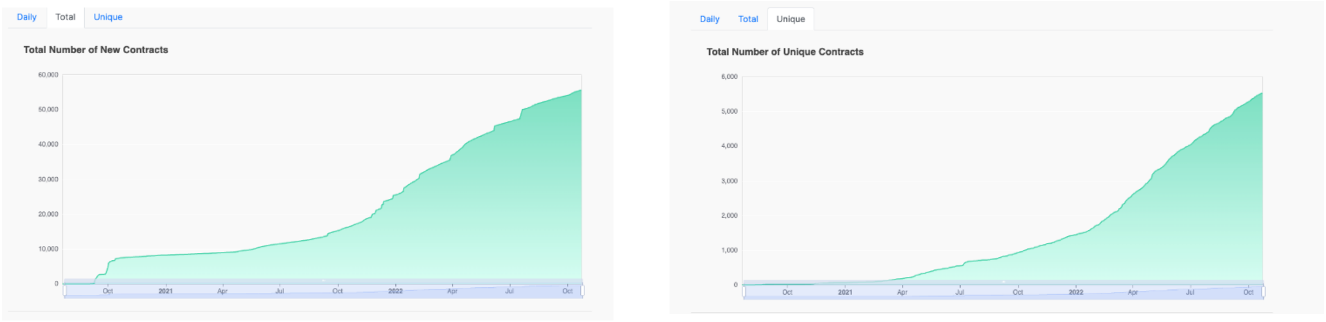


Figure 8: Contracts uploaded on NEAR blockchain, Source: [NEAR Explorer]

In comparison, new blockchain networks such as Aurora, are featuring a small but significant number of new verified smart contracts. The number of new verified smart contracts in Aurora is between zero to 35 contracts on a daily basis.

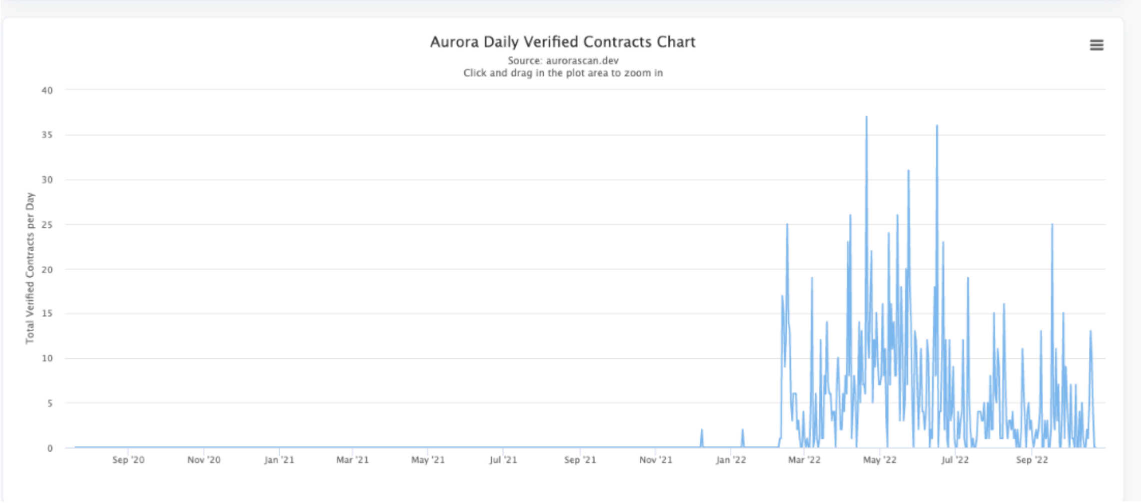


Figure 9: Contracts uploaded on Aurora blockchain with verified source codes only, Source: [Aurora Scan]

A similar pattern is followed on the Optimism blockchain with the total number of new verified smart contracts varying between 9 and 50.

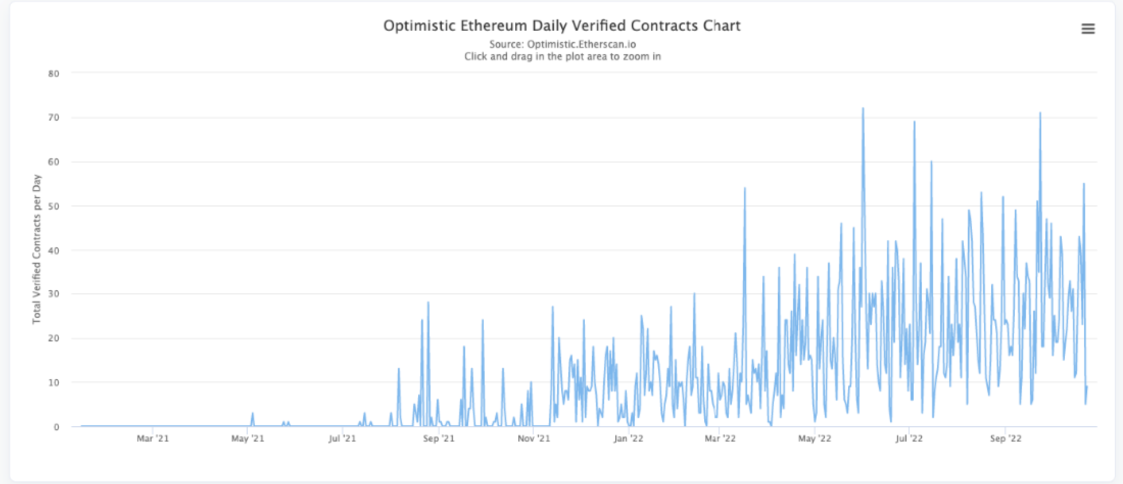
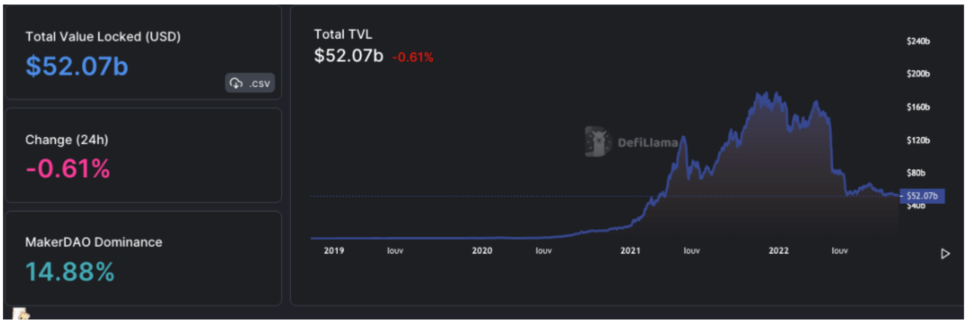


Figure 10:Contracts uploaded on Optimism blockchain, Source: [Optimistic]

Unfortunately, there is no unified metric to extract the number of smart contracts deployed on every single blockchain network, and each block explorer uses a different methodology to track the total, the new, and the verified smart contracts. Analysing the above metrics is not sufficient to provide important insights into the current state of adoption.

Below we will briefly present other data-driven metrics that could help us understand better the adoption of smart contracts.

Total Locked Value:



Total Value Locked (USD)
$52.07b

Change (24h)
-0.61%

MakerDAO Dominance
14.88%

Total TVL
$52.07b -0.61%

Value Locked is an indicator to evaluate the adoption scale of a DeFi project by calculating the total value (USD) of all decentralised finance smart contracts on a particular blockchain. According to data extracted from DefiLlama which tracks 120 blockchain networks, the Total Value Locked across DeFi protocols is surpassing USD 52 billion, following a downwards trend since January 2022.

Developer Activity:

One of the most important metrics to evaluate the adoption of smart contracts is the measurement of developer activity. According to the 2021 Blockchain Developer Report conducted by Electric Capital there were more than 18,000+ monthly active developers who commit code in open source crypto and Web3 projects. A total of 34,000+ new developers committed code in blockchain projects in 2021 — the highest in the history of blockchain. The largest ecosystems are Ethereum, Bitcoin, Polkadot, Cosmos, Solana, BSC, NEAR, Avalanche, Tezos, Polygon, and Cardano, each with 250+ monthly active developers. Among the interesting insights of the report is that more than 60% of the total of blockchain developers only started their blockchain contribution during 2021, demonstrating that 2021 was a critical year for the current state of smart contract adoption. According to historic data from other software segments, only 10% of all devs will retain through the end of year 2.

To understand the state of adoption, we also need to evaluate the total number of active users and the total number of transactions engaging with smart contracts. This report does not aim to go into depth and act as a comparison of different blockchain networks, hence we will not expand into such metrics further, but rather present the current state of smart contract market observations.

The above data collectively can provide useful insights into the current state of adoption of smart contracts.

2.2 Legal Considerations and Paradigm from Regulators Across Europe

2.2.1 The Data Act[4]

Smart Contracts as defined in the Data Act

The European Commission, aiming at building a genuine single market for data and at making Europe a global leader in the data-agile economy has proposed the Data Act as a regulation to harmonise rules on fair access to and use of data. The legislative process in the European Parliament and the Council is currently underway. In particular, "it contributes to the creation of a cross-sectoral governance framework for data access and use by legislating on matters that affect relations between data economy actors, in order to provide incentives for horizontal data sharing across sectors."

In the Commission Proposal for the Data Act (published in February 2022), Smart Contracts are defined as "computer programs on electronic ledgers that execute and settle transactions based on predetermined conditions. They have the potential to provide data holders and data recipients with guarantees that conditions for sharing data are respected. As such, they facilitate the automated and interoperable use of data. The use of electronic ledgers implies they use advanced encryption techniques, and can be decentralised and distributed, resulting in immutability."

Smart contracts have the potential to ensure that conditions for sharing data are respected. Thus, smart contracts are of particular relevance for data transfers and data pooling, since they can give data holders and data recipients trust, that data agreements are followed. The proposed regulation aims to promote the interoperability of smart contracts in data sharing applications. Smart contracts are addressed under Chapter VIII of the Data Act legislative proposal, in Article 28 (1) (d) and Article 29.

With regards to smart contracts, the aim is to promote their interoperability by providing a set of requirements for professionals working on smart contracts for third parties or for those that integrate them in applications that support the implementation of data sharing agreements. There will be a presumption of conformity with the essential requirements for smart contracts that meet harmonised standards or relevant parts of the Standardisation Regulation (No 1025/2012) while in the case that harmonised standards do not exist; the European Commission may take steps to develop and adopt them.

Requirements for Smart Contracts under the Data Act

There are four essential requirements for smart contracts, listed in Article 30 (1) of the Data Act:

"The vendor of an application using smart contracts or, in the absence thereof, the person whose trade, business or profession involves the deployment of smart contracts for others in the context of an agreement to make data available shall comply with the following essential requirements:

> (a) robustness: ensure that the smart contract has been designed to offer a very high degree of robustness to avoid functional errors and to withstand manipulation by third parties;
>
> (b) safe termination and interruption: ensure that a mechanism exists to terminate the continued execution of transactions: the smart contract shall include internal functions which can reset or instruct the contract to stop or interrupt the operation to avoid future (accidental) executions;
>
> (c) data archiving and continuity: foresee, if a smart contract must be terminated or deactivated, a possibility to archive transactional data, the smart contract logic and code to keep the record of the operations performed on the data in the past (auditability); and

(d) access control: a smart contract shall be protected through rigorous access control mechanisms at the governance and smart contract layers."

2.2.2 National approaches

Italian Law on Smart Contracts[5,6]

In February 2019, with Law No 12/2019, the Italian Parliament managed to complete the conversion procedure of Law Decree No 135/2018.

The Italian law defines distributed ledger technology as follows: "Information technologies and protocols that use a shared, distributed, replicable, simultaneously accessible register, architecturally decentralized on cryptographic bases, such as to allow the recording, validation, updating and archiving of data either in clear or further protected by cryptography, verifiable by each participant, not alterable and not editable."

According to this law, the recording of an IT document through the use of DLT produces the legal effects of the electronic time validation referred to in Article 41 of Regulation (EU) no. 910/2014 (eIDAS) according to which:

1. An electronic time stamp shall not be denied legal effect and admissibility as evidence in legal proceedings solely on the grounds that it is in an electronic form or that it does not meet the requirements of the qualified electronic time stamp.

2. A qualified electronic time stamp shall enjoy the presumption of the accuracy of the date and the time it indicates and the integrity of the data to which the date and time are bound.

3. A qualified electronic time stamp issued in one Member State shall be recognised as a qualified electronic time stamp in all Member States."

The second innovation introduced by the law is the recognition of smart contracts and their full validity. This however still requires the publication of the guidelines by the Italian Digital Agency (Agenzia per l'Italia Digitale) – which was supposed to be issued 90 days after the law had passed (i.e. in May 2019); unfortunately it is not yet available. In accordance to the Italian law, the smart contract will need to satisfy the requirement of the written form when requested (similarly to the self-compliance declaration referred in the EU Data Act).

According to Italian law, the smart contracts are: *"A computer program that operates on technologies based on distributed ledgers and whose execution automatically binds two or more parts on the basis of predefined effects. The smart contracts satisfy the requirement of the written form after the IT identification of the interested parties, through a process having the requirements set by the Agency for Digital Italy with guidelines to be adopted within 90 days from the entry into force of the law converting the decree law".*

A complete equivalence between the "traditional" contracts on one hand and the IT documents on the other has been established. Still, for this equivalence to be relevant in all the areas of law, the contracts need to be in written form as required (but not yet defined).

UK Law on Smart Contracts[7]

Although the UK is not considered a favourable jurisdiction for "crypto-enthusiasts", the latest works on the legal and regulatory framework, have established a good ground for the development of blockchain technology based financial service providers. Currently, the sole fact that English law is able to accommodate smart contracts, provides a level of comfort and certainty to the parties of that contract.

The Law Commission in the UK is a statutory independent body aiming to ensure that the law is fair, modern, simple and cost-effective; they conduct research and consultation for providing recommendations to the Parliament and they are responsible for codifying the law, eliminating anomalies. As part of their mandate, the provide guidelines in the form of reports, instructing how the law should be applied.

In the case of smart contracts, more specifically, they “detailed an analysis of the application of existing law to smart contracts, considering a number of scenarios and feedback from industry stakeholders, and the legal community”.

Their recommendation was that the legal framework in England and Wales is able to support and facilitate the use of smart contracts. The report addresses in particular "smart legal contracts". A smart legal contract is a legally binding contract in which some or all of the contractual obligations are defined in and/or performed by the execution of code in a computer programme. It is important to clarify that a smart legal contract is not necessarily programmed on a distributed ledger.

Accordingly, three categories of smart legal contracts were defined: 1) a natural language contract with automatic performance by code; 2) a hybrid contract, where the contractual language is in both natural language terms and in code; and 3) a contract that is exclusively recorded in code.

There are certain issues here that need to be taken into account, such as formation of the contract, interpretation of smart contract content in case of disputes, consumer protection, remedies and courts of jurisdiction.

> **Contract form:** Under English law, there are a number of requirements for a legally binding contract to be formed: agreement (an offer to be bound on specified terms, and acceptance of that offer), consideration, certainty and completeness, intention to create legal relations, and compliance with formalities. The Law Commission concluded that all of those requirements could be met by a smart legal contract.
>
> **Interpretation:** In interpreting contracts which are subject of a dispute between parties, the courts will consider what the language would mean to a reasonable person, with all of the background knowledge available to the parties at the time at which the contract was made. The addition of code into the interpretive mix is likely to give rise to interpretive difficulties. The Law Commission suggests that the test used should be a version of the "traditional" test: what would a person with knowledge and understanding of code understand the coded term to mean? This is, the Commission says, consistent with the existing approach to contractual interpretation.
>
> **Remedies:** There are various issues which might arise in this area, many of which are practical rather than legal. For example, it is not possible to amend a smart legal contract which has been deployed on an immutable distributed ledger, so in order to achieve rectification a court might need to order a party to deploy an amended contract onto the ledger. Parties should ensure, and make provisions for the case that the code for a particular smart legal contract does not perform as anticipated. From a legal perspective, if a smart legal contract is solely in code, it may be particularly difficult to establish a breach of contract due to the interpretive constraints described above, but once interpreted, courts should be able to apply existing principles to determine whether a breach has occurred.
>
> **Consumer protection:** In the case where a smart legal contract is a “business to consumer” ("B2C") contract, consumer protection must be taken into account. This is of particular concern to financial service providers directly providing services to consumers via smart legal contracts. Where a smart legal contract is only coded, or even hybrid, firms will need to ensure that there is a clear natural

language explanation which accompanies the coded contract, and that any consumer right to cease the contract is feasible.

Jurisdiction: Jurisdictional issues may arise in the case of cross-border contracts, where clauses on governing law and/or choice of court are absent; in such cases, standard rules under private international law apply for the purposes of determination which courts have jurisdiction to hear disputes and which governing law shall apply. Given that DLT based systems rely on information processing performed by multiple nodes, located in physical locations that can be spread across different jurisdictions, it is hard to identify a particular physical place of contract formation. The Law Commission has recommended that the parties to a smart legal contract should include appropriate jurisdictional and governing law clauses in their contracts in order to mitigate legal uncertainty that can arise in their absence.

Although the English law provides an interesting approach to smart legal contracts, there is still legal uncertainty, on the above mentioned issues, that do not guarantee a favourable environment for smart contracts deployments.

2.3 Thematic Use Cases

2.3.1 DAOs

2022 has been coined “the year of the DAO” by Messari and therefore, tops this list of smart contract use case applications. DAOs (Decentralized Autonomous Organizations) are a type of organizational governance structure consisting of internet native communities who work with digital infrastructure, and focus on a shared mission and blockchain protocol. They contain incentive based approaches to operating in a free market. Their assets are managed and owned by contributors (members) of the community, and they fluidly operate bottom up with flat, yet still structured, hierarchies. Hence, an immediate question that arises regarding DAOs is which legal jurisdiction applies to it if i.e., it runs on nodes distributed across the globe and involves interactions of citizens of different countries, of which identities not always are perfectly well known to regulatory authorities.

DAOs often operate an open membership format, are open source, and utilize smart contracts for their functioning. This format enables transparency, global accessibility and rights regarding exiting the network. Assets and capital are represented in the format of tokens, which can be used to exercise voting power, govern a protocol, allocate funds, enforce community social norms and incentivize network participation. Aims are focused on marketplace and operations automation, and on collusion prevention. Some examples of what DAOs and their underlying system of smart contracts can manage and govern include: cryptocurrency protocols, decentralized exchanges, NFT collections, pooled labour, charity groups, social clubs, identity, etc.

For example, DAOs could benefit from Trust Anchors approach which is GDPR compliant due to identity data being kept outside of the blockchain, and thus is possible to erase. On the other hand, on the blockchain, the identity data is encrypted which helps Trust Anchor’s operator too. ISO TC 307 is currently preparing a dedicated report on Trust Anchors.

Benefits and advantages offered by DAOs may currently lean towards outweighing the disadvantages. This can largely be seen in labour, income, value capture and by participation in the gig economy. Many experts working in the crypto-space[8],[9],[10],[11],[12] believe that DAOs will be and already are becoming the future of work. DAO contributors can either be full or part time employed, or participate in task-based project work. A unified sense of purpose, flexible and global work from home hours, rewards for early adopters, and low entrance barriers (that often simply begin by joining a Discord server), all make for highly attractive and lucrative approaches to a modern workforce. Regardless of contribution, those participating early in a DAO can retain royalties in connection to a DAO project, or to the organization's ongoing success. This unlocks entirely new approaches to talent sourcing in that the dependencies on geographical borders, educational background and qualifications, and lengthy hierarchical hiring processes begin to dissolve. Also it creates questions such as social security and rights of these employees, as well as labour taxation. Quite an important aspect is how organisations structured as DAOs will decide to move forward with VAT and taxation obligations that might arise from their activities – an interesting paradigm could be the automation of taxation via smart contracts that could perform this function as well following a clearly predefined set of rules.

Entirely new forms of organizational governance are unlocked in DAO set ups that are distributed, automated, highly adaptive and transparent. Flat hierarchies foster decentralized leadership and management where the DAO acts as a managerial layer that programmatically decides, via smart contracts, how to carry out real-world business or charitable operations. This is made possible because of business and/or operational logic in the smart contracts and is updated by the community using governance mechanisms. An example of this can be found in a voting (polling) contract, which is a type of smart contract that allows community participants (token holders) to vote on decisions and proposals using their tokens. While DAOs can be set up for any sector, current industries being impacted by them include: **finance**, **cultural**, **software**, **education** and **professional services**, and respective examples include: MakerDAO, Friends With Benefits, Aragon, DeveloperDAO and LexDAO. In addition, **non-profit work** is similarly being impacted, and can be seen in communities such as KlimaDAO and MolochDAO.

Day-to-day operations and functioning of DAOs currently face numerous challenges. Communication and information flow across communities can be slow, unstructured and requires enormous amounts of filtering to reach the correct community members. Governance challenges are abundant regarding accountability, contributor engagement and decentralized decision making. A balance is to be struck between leadership and decentralization needs, and trends are showing that certain decision making processes may end up swinging back towards how they are done in traditional companies. In addition, the tooling, especially decision support tooling, is underdeveloped and in its infancy. It is considered a critical and existential need amongst crypto communities and is a current investment trend this year. While DAOs are subject to the smart contract security vulnerabilities described in Chapter 1.3., member privacy and agency concerns also exist due to the transparent nature of blockchain recorded actions and other publicly accessible DAO operations related communications.

For DAOs, a very important aspect is the KYC and the identity. DAOs could benefit for example from Trust Anchors approach which is GDPR compliant due to identity data is being kept outside of blockchain, and thus is erasable. On blockchain is kept only cryptographic information which helps Trust Anchor's operator. ISO TC 307 is preparing a dedicated report on Trust Anchors.

When it comes to legal and regulation, we are in very early days. DAOs are currently not categorized as specific legal entities in most parts of the world, which makes for rather opaque regulatory enforcement. Efforts are underway to better understand how DAOs actually work when it comes to contract law and how to address contributor liability. In addition, jurisdictional banking, tax and employment compliance requirements are unclear. Questions remain regarding the applicability of existing laws and corporate entity forms, versus the creation of entirely new and legally recognized forms. Hybrid centralized-decentralized approaches and special corporate legal frameworks for DAOs have been attempted, however it's likely that entirely new and unique approaches may be needed.

2.3.2 NFTs

While DAOs are the stand-out blockchain project from 2022, 2021 was the year of the Non-Fungible Token ("NFT").

A non-fungible token is a unit of ownership of an asset (i.e., a token) which is not interchangeable (i.e., non-fungible). In other words, an NFT is a unique digital asset. Contrast this with the fungibility of fiat money - money by definition is interchangeable.. Similarly, identical photocopies of an image can also be said to be interchangeable NFTs are not interchangeable because the smart contract embedded within the NFT can differentiate between the different (yet identical) assets.

The first NFTs were built on the Bitcoin blockchain in 2012. The earliest NFT assets included in-game currencies such as BitCrystals, cardgame-trading and meme-trading.

From 2017 onwards, memes began to be traded on the Ethereum Blockchain. "Cryptopunks" was launched in 2017. This was the creation of 10,000 unique characters. Later that year, the blockchain based virtual game 'CryptoKitties' was released. The launch happily coincided with the 2017 crypto bullrun. Investors were investing heavily in crypto and blockchain assets, and NFTs 'had arrived'.

The Ethereum Blockchain developed a set of technical standards applicable to the different types of activities and tokens deployed on the network. These standards ensured a cohesive approach to token development and enabled interoperability and compatibility between different tokens and applications. ERC20 was initially the most common token standard. ERC721 was developed specifically for NFTs. Its smart contract incorporates more complex code which allows the NFT's ownership and trading activities to be tracked and recorded. Early platforms which 'mint' NFTs include Rarible and OpenSea. Now, there are dozens.

2.3.3 Identity Credentials & Data Management[13]

Quite a number of apps have been developed since the beginning of COVID-19 and especially since the testing and vaccination process was put in place, aimed at using verifiable credentials to simplify the transfer of data (in the COVID-19 testing process). An interesting use case is the one developed by Bart Cant, Founder and Managing Partner of Rethink Ledgers, who used a unique approach for his prototype. The app was developed to use verifiable credentials for secure sharing of COVID-19 test results and vaccine information, and was also based on smart contracts.

The app is titled "State Surveillance System for Covid-19 Testing and Vaccine Distribution Management". It was developed using DAML (Digital Assets Modeling Language) and W3C's verifiable credentials; it offers a secure digital experience for citizens, health clinic providers, and state agencies, sharing COVID-19 test results, "proof of vaccine" administration, and other "immunity proofs" using a centralized ledger.

In a recent interview, Bart Cant explained that the app was built to address a number of challenges with regards to provision of 1) a secure, privacy-enhanced and safe environment for communicating testing results and 2) accuracy and timeliness of data to support policy setting (i.e., sharing data with state and/or local agencies in an automated and secure manner).

The app is built using a combination of the DAML privacy-enhancing, smart contract language and W3C's verifiable credentials. DAML guarantees privacy by coding the solution as rights and obligations. The information is stored, so it cannot be accessed unless the party accessing that data has been explicitly configured in the smart contracts to access it. DAML is also one of the few solutions that allows interoperability across networks. The solution is built on a centralized ledger based solution (https://projectdabl.com/) but could be easily deployed on other blockchain networks (e.g., Hyperledger Fabric, Sawtooth, Besu or R3 Corda), centralized ledgers (e.g., AWS QLDB), or traditional databases (e.g., Postgres). The major technical component of the solution is how it interacts with verifiable credentials. Verifiable- credentials are used to securely transmit COVID-19 test information from the health clinics to the patient. The application provides a digital workflow for this experience (including a mobile experience) while still being HIPAA (Health Insurance Portability and Accountability Act)[14] compliant and providing a safe and secure mechanism for sharing COVID-19 test data that allows the recipient to be in control of their data and decide on what to share and with whom.

Since the COVID-19 pandemic, there has been a great emphasis on privacy and regulation. Verifiable credentials allow sharing medical data directly from the issuer to the holder without the need to store the data. Smart contracts can be leveraged to orchestrate this process. Additionally, smart contracts can be used to anonymize data by using DIDs, for instance. Additionally, the critical medical data can be blinded and encrypted by the DAML smart contract, so the solution can be both HIPAA and GDPR compliant.

2.3.4 Decentralized Finance

Decentralized Finance (DeFi) is an umbrella term for a collection of open and peer-to-peer financial products that rely on blockchains and smart contracts. DeFi protocols are non-custodial, and are known for their public and transparent operation since the transactions are publicly available. The DeFi economy involves in a high degree unregulated actors, which are not supervised by a national authority. DeFi protocols are inclusive to anybody who is interested in transacting. The DeFi protocols are highly radical and disruptive and promise to make financial services accessible to everyone. The current Total Value Locked – a key metric to measure the size of the Decentralized Finance economy is exceeding $54 billion, following a significant downtrend since January 2022 and the peak of $178 billion. The biggest protocol is the MakerDao, a smart contract lending platform that enables users to take out loans by locking-in collateral in exchange for Dai which has a market dominance of 14.5%. Currently the total number of Decentralized Finance protocols is exceeding 2000.

There is an extensive differentiation between the different DeFi categories and use cases. According to data extracted by DeFi Lama, we can breakdown the use cases in 28 categories. In terms of market adoption and size, the leading use case are the Decentralized Exchanges, non-custodial protocols that allow users to swap or trade cryptocurrencies. The second most popular use case are the decentralized Lending Protocols that allow users to borrow or lend assets. Other use cases include the CDP- protocols that mint their own stablecoins using collateralized lending, Bridges that allow you to transfer tokens from one blockchain to another, and liquid staking protocols that allow you to stake your assets in exchange of a yield reward, along enabling tradeable and liquid staking positions. Other use cases include Yield Generation Protocols, Derivative Markets, Synthetic Assets, Insurance Protocols, CrowdFunding and Launchpads, Tokenised Indexes, Payments, NFT Lending, Undercollateralized licensing, and Prediction markets.

According to the European Blockchain Observatory & Forum report about the Decentralized Finance Market DeFi promises to be one of the most disruptive applications of blockchains and smart contracts. The experts foresee that the current landscape of DeFi applications in existence, is only the tip of the iceberg compared to the expected innovation the DeFi market will deliver in the near future. Currently the real impact of DeFi has minimal impact on the real economy and the use cases are artificially limited in the cryptocurrency markets. In a European level, the decentralized finance maret is expected to contribute to the competitiveness of the European economy, along introducing new forms of financing for small and medium-sized enterprises, and European citizens, along contributing to Europe’s ambitions to make financial services more accessible and transparent.

2.3.5 Synthetic Tokens & Indexes

An interesting application of smart contracts is the creation of **Synthetic tokens** that created tokenized derivative that mimics the value of another asset, and **Indexes** that track the performance of a group of related assets.

Synthetic assets constitute Decentralized Finance analogues of derivatives that onboard the traditional finance (TradFi) and enable novel DeFi use cases that allow investors to derive value from the fluctuations in the value of an underlying asset. Synthetic assets use oracles to track the price of an underlying asset – with cryptocurrencies, stocks, and physical commodities being among the most popular categories. Synthetic assets could be summarised as tokenized derivatives that feature reduced reliance on middlemen, modularity and fractional ownership, reduced geographical barriers, and expanded asset liquidity. While synethtic assets are a fundamental block for the growth of the DeFi landscape, it is an asset class that is more prone to the regulatory supervision. Currently the Total Locked Value (TLV) of Synthetics protocols is exceeding $643 million with 27 decentralized finance protocols building in this market. The three biggest protocols as measured in TLV are the Synthetix, Injective, and Alchemy. Because synthetic assets are tokens, you can utilise them on other DeFi platforms, provide liquidity to markets, and earn interest. Synthetic assets as new primitives are prone to risks such as smart contract, governance, oracle and blockchain network risks and inefficiencies.

Tokenised indexes are decentralized finance protocols that track the performance of a group of related assets. There are currently 32 notable protocols, and their market share is approximately $240 million. The three leading protocols in terms of TLV are the Set Protocol, Index Coop, and Enzyme Finance. The most widespread application is the tokenized index funds which are structured in a similar manner with the index funds. The tokenized indexes provide with thematic exposure to particular asset classes. The assets are locked in a smart contract, allowing the tokenized indexes to follow their pegged assets.

2.3.6 Renewable Energy tokens[15]

The **WePower** platform brings together Renewable Energy Sources generators and investors interested in supporting sustainable energy projects. Renewable energy produced is tokenised and subsequently traded through the platform either to purchase electricity or exchanged for fiat currencies or cryptocurrencies. The platform uses **blockchains** and **smart contracts**.

WePower has tokenized – turning it into thirty-nine billion smart energy tokens and uploading it on a blockchain - a year's worth of Estonian hourly production and consumption data. The hourly data from 700,000 households was aggregated by postal code, per hour, to preserve privacy and reduce data to a manageable size.

Estonia was chosen because it has 100 percent smart meter coverage and a smart meter data platform (called Estfeed) to provide detailed data. Each token is essentially a digital self-settling power-purchase contract representing one kilowatt-hour of power. The tokens are tradeable and can be sold into the local energy wholesale market by linking the digital contracts with power grid data on the blockchain.

The financial and legal instruments that currently exist in the market, such as Power Purchase Agreements (PPAs), are too complex, lengthy and expensive for energy buyers. They aim to enable all companies, regardless of their size, to enter this market.

Lack of scalability is one of the biggest roadblocks Wepower met: While Ethereum is currently one of the most mature blockchain solutions supporting smart contracts, large-scale energy trading on the blockchain is not yet fully feasible. Wepower aims to keep on working on different technologies while monitoring the development of Ethereum and other blockchains.

2.3.7 Enforcing/Billing Usage Agreements

A critical aspect when implementing solutions based on Smart Contracts has to do with the purity of the information that the latter receive. In this way, the less intermediaries participate in the transfer of information to the Blockchain, the more reliable the results processed in the Blockchain will be.

A clear example of this approach is the one documented at https://wso2.com/library/conference/2018/11/wso2con-eu-2018-blockchain-in-the-business-api-ecosystem/.

The consumption of APIs is a concept that is clearly defined (conditions, costs and billing). This type of use case is a notable example of the usefulness of Smart Contracts as they can take contractual conditions to a Smart Contract-based solution and execute them in a decentralized environment.

The entry point for Smart Contracts are the billing modules of the API manager itself. From there, the specific rules of the contracts are evaluated within the secure environment of the Blockchain. At the end of the process, an invoice generated through a transparent and non-tamperable process is obtained by the end user.

2.3.8 Emerging Technologies[16]

One of the most exciting applications of blockchain technology and associated smart contract technology is the ability to facilitate complex computational tasks like those involved in machine learning (ML) and artificial intelligence (AI). By combining the data intensive processing of AI with the decentralized security and immutability of blockchain technology, there is potential to create AI-powered smart contracts. As smart contract applications become implemented across various industries, they will need to become increasingly complex to accommodate their new roles. While fundamental smart contract use cases can be manually designed, AI-enabled smart contracts might allow for the construction of highly complex, more responsive, enterprise-grade smart contracts and DApps that have the potential to dramatically expand the capabilities of the technology.

Several experts in this area suggest that the fields of AI and blockchain may benefit from each other's defining characteristics. Smart contracts can benefit from the advanced computational capabilities and adaptive systems of AI technology, while AI implementations could utilise smart contract technology for its autonomous execution of sets of rules and to provide a secure environment for sensitive and valuable machine learning data to exist. Zilliqa is one of the many blockchain platforms that is developing advanced computational capabilities with its proprietary smart contract programming language, Scilla, and advanced parallel processing structure enabled by sharding.

Chapter 3: Highlights and Conclusions

The term “smart contract” was originally created by Nick Szabo, a digital scientist and cryptographer mainly known for his research in digital contracts and digital currencies, as well as his invention of a virtual currency called “Bit Gold” in 1998. In 1994, he published a book, titled “Smart Contracts: Building Blocks for Digital Free Markets**”,** where he introduced the notion of smart contracts. At the time blockchain technology did not exist, so his ideas were not tested.

In his book, Szabo described Smart Contracts as tools that “automate the execution of agreements, and ensure that all participants can view the outcome as quickly as possible, without the involvement of an intermediary.”

But how do they work? Smart contracts are self-executing contracts in which the buyer and seller agreements are documented and embedded directly into lines of code. The adoption of smart contracts serves to make transactions traceable, transparent, and irreversible; they serve as the backbone of the Web 3.0 ecosystem, and enable users to interact online, leveraging blockchain as the main driver.

The key characteristics, which could also be considered as the main benefits of using smart contracts hosted by blockchain ecosystems are:

- They are trustless and reliable: Smart contracts always execute as programmed. No third-party risk means users can be certain of the expected outcome as per the pre-defined code.
- They offer enhanced security: The code lives within the immutable ledger distributed across the entire network, significantly limiting any single point of failure. After the contract is deployed, the blockchain provides integrity and security.
- They cost less: The absence of third parties removes overhead costs, bringing efficiency to smart contracts.
- They are faster: Transaction finality is almost instant, depending on the blockchains’ block times and rules.

A smart contract provides a more efficient and enhanced mechanism of agreement, transacting, and transferring value. Furthermore, smart contracts are capable of executing any special-purpose task, so they are the foundation of a potential DApp movement.

What is more important with smart contracts, is their programmable logic, which is applicable in any traditional sector and can support the implementation of many other applications. These may range from on-chain data access and storage to cybersecurity, voting management systems, healthcare data management, etc.

On the other hand, there seem to be two primary concerns and downsides that will have to be addressed in order for smart contracts to be more widely adopted:

First, for a contract to be inserted onto the blockchain, all of the necessary legal jargon has to be translated into computer code. As legal professionals are not usually coders themselves and coders normally don’t have a legal background, a certain level of trust as well as expertise need to be maintained, to ensure that all parties can trust that the code within the smart contract truly reflects the legal content and purpose.

Second, a blockchain-based smart contract is “written in stone”, i.e., cannot be altered. The blockchain is decentralized, and usually that is a good thing. **But** it also means that no central authority or referee will be able to step in, in case one party feels wronged or even defrauded.

It is expected that the above issues will be resolved eventually. As more and more smart contracts are drawn up, they shall serve as examples or templates for similar agreements. Current contract disputes may require an attorney or a small claims court session. It is likely that these kinds of service contracts will include some sort of financial guarantee in the future - a deposit, for example, can easily enable this agreement to be managed, enforced, and policed via a smart contract.[17]

It is likely that the EU will rely on harmonised requirements and standards to facilitate the use of smart contracts if the current smart contract provisions are approved in more or less their current form. The EC is taking a different stance than the UK in terms of the need to legislate for enabling the effective adoption of smart contracts. UK appears to be on the more flexible and adoptive side, but still, it seems that businesses entering into smart contracts with EU counterparties, will want to align with Article 30 Data Act requirements, even if the contracts are governed by English law.[18]

References

Allam, Z. (2018). On smart contracts and organisational performance: A review of smart contracts through the blockchain technology. Review of Economic and Business Studies, 11(2), 137-156.

Altay, H., & Motawa, I. (2020). An investigation on the applicability of smart contracts in the construction industry. In Workshop Proceedings (p. 12).

Chainlink. (Last updated: July 22, 2022)/ Smart Contracts Introduction. https://chain.link/education/smart-contracts#smart-contract-limitations.

Cointelegraph. (October 02, 2022). Transit Swap loses over $21M due to code bug exploit, issues apology. https://cointelegraph.com/news/transit-swap-loses-over-21m-due-to-internal-bug-hack-issues-apology. Accessed on:

Crypto News. (September 21, 2022). Ethereum Scaling Solution Aurora Pays $2 Million Bug Bounty to Hackers. https://cryptonews.com/news/ethereum-scaling-solution-aurora-pays-2-million-bug-bounty-hackers.htm. Accessed on:

Decrypt. (03 October, 2022). Blockchain Builders Need Bug Bounty Programs: Immunefi Engineer. https://decrypt.co/111142/blockchain-builders-need-bug-bounty-programs-immunefi-engineer. Access on:

Ethereum Blog. (Last edited: September 2, 2022). Introduction to smart contracts. https://ethereum.org/en/developers/docs/smart-contracts/#limitations.

Ethereum Improvement Proposals. (2016). EIP-170: Contract code size limit. https://eips.ethereum.org/EIPS/eip-170.

Ethereum Improvement Proposals. (2020). EIP-2535: Diamonds, Multi-Facet Proxy. https://eips.ethereum.org/EIPS/eip-2535.

Gemini. (October 6, 2022). The DAO: What was the DAO Hack?. https://www.gemini.com/cryptopedia/the-dao-hack-makerdao#section-the-dao-hack. Accessed on:

Gitlab. Ethereum Smart Contract Best Practices. https://consensys.github.io/smart-contract-best-practices/security-tools/. Accessed on:

Hedera. Hedera bug bounty program. https://hedera.com/bounty. Accessed on:

Kroll, J. A. (2015). Accountable algorithms (Doctoral dissertation, Princeton University).

Luu, L., Chu, D. H., Olickel, H., Saxena, P., & Hobor, A. (2016, October). Making smart contracts smarter. In Proceedings of the 2016 ACM SIGSAC conference on computer and communications security (pp. 254-269).

Nehai, Z., & Bobot, F. (2019). Deductive proof of ethereum smart contracts using why3. arXiv preprint arXiv:1904.11281.

Nzuva, S. (2019). Smart contracts implementation, applications, benefits, and limitations. School of Computing and Information Technology, Jomo Kenyatta University of Agriculture and Technology, Nairobi, Kenya.

Sánchez, D. C. (2018). Raziel: Private and verifiable smart contracts on blockchains. arXiv preprint arXiv:1807.09484.

Web Sources:

https://www.merriam-webster.com/dictionary/NFT

https://medium.com/@Andrew.Steinwold/the-history-of-non-fungible-tokens-nfts-f362ca57ae10

https://ethereum.org/en/developers/docs/standards/tokens/erc-20/

https://www.lawcom.gov.uk/law-commission-proposes-reforms-for-digital-assets-including-crypto-tokens-and-nfts/

https://moralis.io/erc721-contract-exploring-erc721-smart-contracts/

https://www.lawcom.gov.uk/about/

https://www.taylorwessing.com/en/insights-and-events/insights/2022/03/why-is-english-law-the-smart-choice-for-governing-smart-contacts

https://www.lancepartners.com/italy-recognizes-blockchain-and-smart-contracts/

https://www2.deloitte.com/content/dam/Deloitte/global/Documents/Tax/dttl-tax-global-blockchain-wp-march-2018.pdf

https://medium.com/visionary-hub/smart-contracts-are-the-future-962007fcb276

https://futuristspeaker.com/future-trends/smart-contracts-are-here-and-getting-smarter/

https://chain.link/use-cases

https://hedera.com/learning/smart-contracts/smart-contract-use-cases

https://blog.chain.link/smart-contract-use-cases/

https://trinsic.id/verifiable-credentials-and-smart-contracts-for-covid19-data-management/

https://www.legalzoom.com/articles/what-are-smart-contracts-and-how-are-they-regulated-and-enforced

Endnotes

[1] Source: https://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/smart.contracts.html

[2] Source: https://policyreview.info/glossary/smart-contracts

[3] Source: https://www.youtube.com/watch?v=TDGq4aeevgY

[4] Source: https://digital-strategy.ec.europa.eu/en/policies/data-act

[5] Source: https://afge.legal/media/articoli/priv-cyb/smart-contract-legalmente-utilizzabili

[6] Source: https://www.lancepartners.com/italy-recognizes-blockchain-and-smart-contracts/

[7] Source: https://www.taylorwessing.com/en/interface/2022/smart-contracts/smart-contracts-in-the-data-act#:~:text='Smart%20contracts'%20are%20defined%20in,for%20sharing%20data%20are%20respected.

[8] https://future.com/the-future-of-work-daos-crypto-networks/

[9] https://www.youtube.com/watch?v=dWSkL0jeTac

[10] https://medium.com/vertexventures/the-way-of-the-dao-could-be-the-future-of-work-5ca7f6c8e310

[11] https://www.coindesk.com/business/2021/09/26/daos-may-be-the-future-of-work-but-dont-bet-on-them-being-the-next-big-asset-class/

[12] https://www.weforum.org/agenda/2022/07/web-3-change-the-future-of-work-decentralized-autonomous-organizations/

[13] Source: https://trinsic.id/verifiable-credentials-and-smart-contracts-for-covid19-data-management/

[14] Source: https://www.cdc.gov/phlp/publications/topic/hipaa.html

[15] Source https://investinestonia.com/wepower-is-the-first-blockchain-firm-to-tokenize-an-entire-grid/

[16] Source: https://www.gemini.com/cryptopedia/smart-contract-examples-smart-contract-use-cases#section-applications-of-smart-contracts-in-emerging-technology

[17] Source: https://futuristspeaker.com/future-trends/smart-contracts-are-here-and-getting-smarter/

[18]Source: https://seekingalpha.com/article/4520669-exploring-the-disruptive-potential-of-smart-contracts