

# Blockchain and Smart Contracts: Challenges, Applications, Architecture and Future Trends

Nishant Chaudhari<sup>1</sup>, Aashish Vivek Bhat<sup>2</sup>, Rudra Kabadi<sup>3</sup>, Priya Shelke<sup>4</sup> and Riddhi Mirajkar<sup>5</sup>

<sup>1-5</sup>Vishwakarma Institute of Information Technology, Pune, India

Email: Chaudhari.21910683@viit.ac.in, {aashish.21911060, rudra.21911008, priya.shelke, riddhi.mirajkar}@viit.ac.in

**Abstract**—Applications in the field of blockchain and cryptocurrencies have gradually expanded as a result of a significant rise in the acceptability of blockchain over the past 10 years. Blockchain-related smart contracts have opened up new avenues for creativity and technical advancement. IoT, management, financial services, the digital economy, and the intelligence sectors all can bring use of smart contracts in various ways. Well-known blockchain-based development platforms like Ethereum and Hyper ledger have also adopted smart contracts. Since smart contracts are still in their early stages of development, serious technical issues including privacy and security concerns need to be looked into further. In this article, we concentrate on giving readers a glimpse inside the blockchain industry while also attempting to further the potential of smart contracts through our study. Additionally, we explore the obstacles to this field's advancement and present some architectural breakthroughs for smart contracts. Overall, we give a survey on the potential uses and difficulties of blockchain technology and smart contracts.

**Index Terms**— Blockchain, Smart Contracts, Ethereum, Bitcoin, Solidity.

## I. INTRODUCTION

It was in the year 2008 that, Satoshi Nakamoto explained the concept of peer-to-peer transfer of money or say cash transactions to the world without the help of interference of central authority [1]. Even though the blockchain was introduced back then, it gained its popularity in the recent years with great upscale in the acceptability of the Blockchain. Blockchain can be stated as the distributed ledger, which is itself being managed by the network nodes, recording the transactions that have been executed among the nodes of a chain. The access of this information is provided to the public, hence it cannot be changed or tempered [5].

Figure 1 shows how the blocks in a block chain are joined together using the hash of previous block that is stored in the next block.

Szabo, a computer scientist, and cryptographer first used the term "smart contract" in the middle of the 1990s. He defined it as "a set of promises, specified in digital form, including procedures within which the parties perform on these promises [3]. Smart contracts are said to be self-enforcing and self-executing programs, written in order to actuate the terms and conditions of a specific document or contract [2]. As the smart contracts are concerned, they operate on decentralized programs on the block chain. This program is immutable so that there can be no issues with the security and to ensure the trust on the programs.

With the introduction of smart contracts, it has become really helpful to make the chain abide by a certain set of rules that every block of a blockchain has to follow [4]. Although blockchains and smart contracts have a tons of

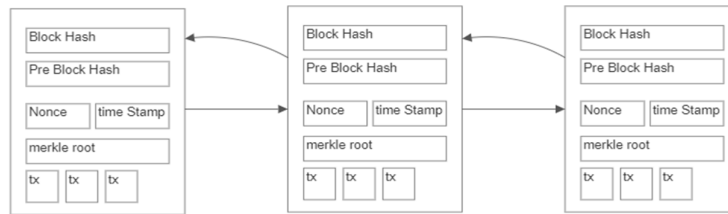


Figure 1. Blocks in a block Chain [1]

benefits, we will also learn that they have a number of drawbacks [6]. The goal of this work is to bring to highlight a descriptive way of how blockchains and smart contacts function in this real world, discuss their pros and cons and further we will have a look at how the introduction of this systems could bring applicative advancements to various fields. Figure 2 explains the functionality of a smart contract on how it responds to the actions on a block chain.

## II. LITERATURE SURVEY

With a recent boom in the area of blockchain, there has been a lot of research out coming in this field. New aspects of this field are arising every day with newer approaches and acceptance is various fields. Here, we would try to bring along some of the new applications of blockchain in various fields, the challenges this industry is facing in the current scenario and discuss some of the architectural advancements in the development of smart contracts and blockchain.

Blockchain technology has undergone recent technical advancements that have made it possible to switch from automatic contracts to genuinely autonomous smart contracts that can execute and enforce themselves [8]. The link between blockchain-based smart contracts and contract law presents an intriguing study setting where the traditional notion of contracts is put under scrutiny as programmed programs start to manage transactions [10]. Smart contracts and blockchain, also known as distributed ledger technology, have a wide range of possible applications. They may be used to improve international payments, syndicated loans, collateral management, proxy voting, securities issuance, and regulatory and compliance processes, according to the World Economic Forum[10].

Whenever a new technology comes into existence with it comes the challenges it faces to encounter the market, its needs and the facilities it can provide to the market. Blockchain and specially smart contracts are no different. Some of the challenges faced by smart contracts are further discussed in the paper.

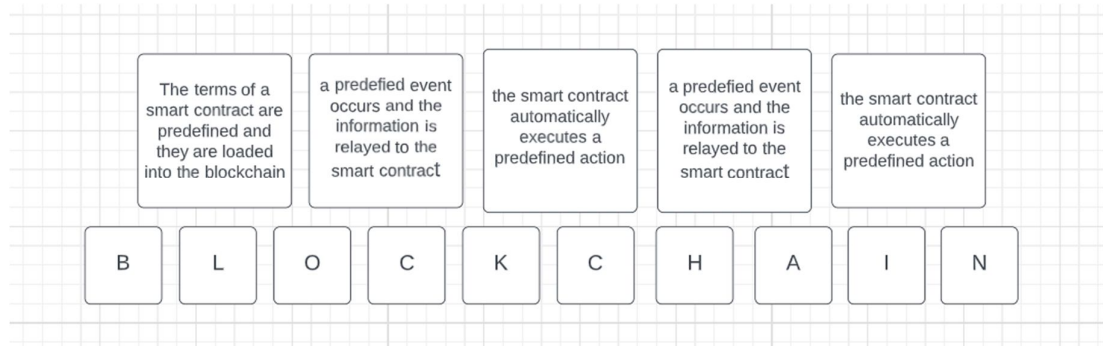


Figure 2. Functionality of a smart contract.[9]

### A. Challenges

A smart contract is a promising technology, but there are still many obstacles to overcome. We divide these major problems into four categories in accordance with the four phases of the life cycle of intelligent contracts. Meanwhile, we also provide a summary of recent developments in resolving those challenging situations [7]. Figure 3 shows the challenges faced by smart contracts.

### Creation challenges

To enact intelligent contracts, contract generation is a necessary step. After implementing their contracts in various blockchain platforms, customers must code their contracts. Blockchain-based completely smart contracts cannot be changed after they have been deployed since blockchains are essentially immutable [12].

### Clarity problems:

Since a smart contract is written in a programming language it makes it nearly impossible for a normal user to read and understand the code. Therefore, to create programs readable in each forms stays a huge aspect to conquer.

Advances to counter readability challenge:

We suggest a reverse engineering tool (called Erays) to investigate built intelligent contracts to address this problem. This contrarian engineering tool can translate hex-encoded contracts into pseudocodes that are understandable to humans [7].

### Functional problems:

Existing smart settlement platforms have a number of practical issues. We show a variety of tough circumstances that are representative:

Re-entrance: Re-entrance denotes the ability to precisely recall the interrupted characteristic once more. Additionally, malicious individuals may utilize this flaw to steal digital currency, as described [12].

Overcharging: It has been demonstrated in recent paintings that smart contracts might be overcharged because of their under-optimization. These overpriced designs have features like lifeless code, expensive loop operations, and repeated computations [12].

Advances for functional issues:

Re-entrance: Several current approaches make an effort to address some of the issues mentioned above. Attacks by re-entrants and issues with money leakage were proposed by Obsidian. Obsidian specifically uses named states to enable consistency checking on nation transitions and verification to allow for the repair of the reentrancy issue [12].

Overcharging: Smart contract developers should be concerned about their execution fees in addition to the effectiveness of their packages. According to reports, gas expensive Ethereum design patterns affect more than 90% of real smart contracts. Additionally, we put forth Gas Reducer, a tool for finding gas-expensive fashions. Green bytecode can take the place of under-optimized bytecode in Gas Reducer [12].

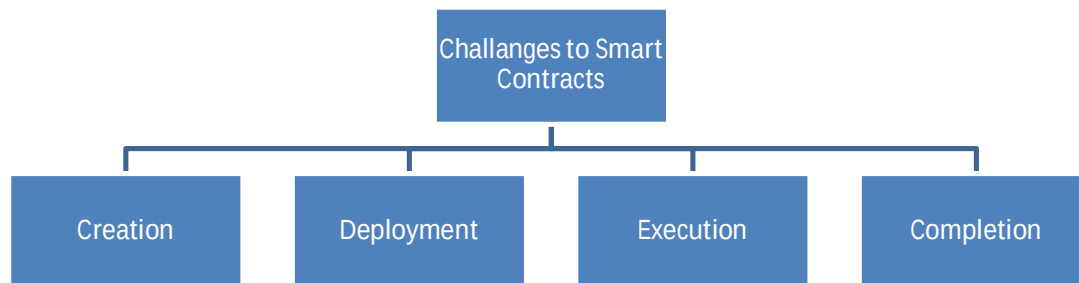


Figure 3. Challenges to Smart contract

### Deployment challenges

Smart contracts may be implemented on blockchain systems after their introduction. However, smart contracts need to be carefully examined to avoid any potential vulnerabilities. Furthermore, astute contract developers seek to minimize ability losses brought on by harmful activities by being mindful of the agreement's interaction patterns. The challenging scenarios and developments in smart contract deployment are then described [13].

*Contract correctness:* It is extremely difficult to make any changes to smart contracts once they have been put on blockchains. Therefore, it is very important to assess the accuracy of smart contracts before their official deployment. The intricacy of modelling smart contracts, however, makes it much more difficult to confirm that they are valid.

### Execution challenges

The execution section of smart contracts is essential since it establishes the final nation for smart contracts. Several issues with the execution of smart contracts must be resolved at some point.

*Reliable oracle:* Real-world data is a requirement for smart contracts to function. An external network is kept separate from a smart contract's sandbox where it can operate. An oracle acts as an agent for a smart contract by uncovering and verifying real-world events and sending this knowledge to the smart contract. Consequently, finding a reliable oracle becomes a challenge [16].

*Dependency on transaction ordering:* Users send transactions to call functions in a smart contract at the same time that miners present the transactions into blocks. Due to the unpredictability of the split blockchain branches, the order of transactions is not deterministic. Order-based transactions may become inconsistent as a result of this ambiguity. This contradiction has been well resolved in conventional database management systems (DBMS), but it is far more challenging to resolve in smart contracts, as we are aware [19].

*Efficiency of Execution:* Miners are used to serially execute smart contracts. A miner will not sign any additional agreements until the cutting-edge settlement is complete, to put it another way. The machine's overall performance is fundamentally constrained by the execution serialization. Due to the shared data across several smart contracts, it is much more difficult to execute smart contracts at the same time. In the meantime, learning how to view the contract details without using a predetermined interface is crucial to enhancing the effectiveness of smart contract execution since it eliminates the need to deploy a new contract [16].

### Completion Challenges

The modification of the device's states following the execution of the smart contract may be packaged as a transaction and broadcast to all nodes. But the rise of smart contracts raises new concerns.

*Security and privacy:* The majority of cutting-edge smart contracts and platforms do not have privacy-preserving features, especially for transactional privacy. In particular, the transaction data are shared across the whole blockchain network. Consequently, everyone within the networks may see every transaction. Even though certain blockchain topologies use pseudonymous public keys to increase transaction privacy, the majority of transaction records (along with balances) are still made available to the public.

Blockchain and smart contracts are vulnerable to harmful attacks brought on by scammers since they are a relatively new technology. For settlement customers in particular, the detection of scams is of utmost importance because it enables them to exit their investments at an early stage to avoid needless loss. Table 1 shows cases more issues that smart contract faces post completion.

## III. APPLICATIONS AND FUTURE ADVANCEMENTS

### A. Blockchain Applications in the field of Insurance

Several enthusiasts have already suggested applying this technology in a variety of fields and situations, such as:

- Government, to transparently record electorate votes, political programs (to check whether promises made have been followed or not), or to enable autonomous governance system [5].
- In the area of intellectual property and rights to approve a certain document and give the end user proof of existence.

TABLES I. ISSUES FACED BY SMART CONTRACTS

| Smart Contract Issues |                                                                  | Proposed solutions                                                                                                                                                                                          |
|-----------------------|------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Codifying issues      | Difficulty of writing correct smart contract [12]                | <ul style="list-style-type: none"> <li>• Semi-automation of smart contract creation [12].</li> <li>• Use of formal verification methods. [12].</li> <li>• Education (e.g., online tutorials)[12]</li> </ul> |
|                       | Inability to modify or terminate smart contracts [12].           | • A set of standards for modifying terminating smart contracts[12].                                                                                                                                         |
|                       | Lack of support to identify under optimised smart contract [17]. | • Use of 'GASPER' tool [16].                                                                                                                                                                                |
|                       | Complexity of programming languages [17].                        | • Use of logic-based languages [17].                                                                                                                                                                        |
| Security issues       | Transaction-ordering dependency vulnerability [18,19].           | <ul style="list-style-type: none"> <li>• Use of 'SendReceived' function [18].</li> <li>• Use of guard condition [19].</li> <li>• Use of 'OYENTE' tool [19].</li> </ul>                                      |
|                       | Timestamp dependency vulnerability [19].                         | <ul style="list-style-type: none"> <li>• Use block number as a random seed instead of using timestamp [23].</li> <li>• Use of 'OYENTE' tool [19].</li> </ul>                                                |
|                       | Mishandled exception vulnerability [19].                         | <ul style="list-style-type: none"> <li>• Check the returned value [19].</li> <li>• Use of 'OYENTE' tool [19].</li> </ul>                                                                                    |
|                       | Re-entrancy vulnerability [19].                                  | • Use of 'OYENTE' tool [19].                                                                                                                                                                                |
|                       | Criminal smart contract activities [19].                         | • N.A.                                                                                                                                                                                                      |
|                       | Lack of trustworthy data feeds 'Oracles' [18].                   | • Use of 'Town Crier (TC)' tool [18].                                                                                                                                                                       |
|                       | Lack of transactional privacy [17].                              | <ul style="list-style-type: none"> <li>• Use of 'Hawk' tool [18].</li> <li>• Use of encryption techniques [18]</li> </ul>                                                                                   |
|                       |                                                                  |                                                                                                                                                                                                             |

- The financial sector can make use of technology to stop depending on banks for user-to-user transactions [11].
- record the qualities and ownership of products, particularly for luxury goods, to cut down on the market for fake and stolen goods.
- Smart contracts can be used by the Internet of Things to interpret sensor data automatically, enabling intelligent machines to communicate with one another and act on their own when certain circumstances arise [6].

#### Financial Applications

Smart contracts' self-executory, irreversible, and distributed properties revolutionize the financial sector in several ways by resolving numerous problems. Smart contracts ensure that the given operation will be carried out on a specific system state and will be carried out without any errors. Highlights the competitive advantages blockchain-based smart contracts provide for financial organizations [2].

#### IV. ARCHITECTURE

In the current situation, smart contracts' fundamental mechanism often has a value and a state characteristic. If-then, "if-else," and other conditional statements are used to build up the trigger conditions and the required answers for the contract terms' correct implementation.

The contract is transferred to the blockchain network after it has been signed by all parties involved. Transactions are broadcast using a P2P network, and several miners then verify the assigned transaction to produce a finalized result that is then recorded in the designated block of the blockchain [7].

While consumers send a transaction to invoke a contract, contract makers receive the returned data throughout the process, such as the contract address [6]. By using their own resources, such as compute, to execute contract code in their own Sandboxed Execution Environment or Ether Virtual Machine , miners participating in the network are attracted to the system's incentive mechanisms, such as crypto coins, prizes, and so on.

If the contract satisfies the triggering conditions for the current situation, it is then evaluated based on the system's current status and input data from reliable sources [14]. If the prerequisites are satisfied, the necessary replies or actions are carried out. After being successfully validated by the entire network's consensus, the information is then put in a new block connected to the entire blockchain.

Ethereum and Hyper ledger Fabric are two examples of this process in action, with the main distinction being that one operates as a public blockchain while the other is built on consortium blockchain. The figure 5 here shows us the better modelling for the current methods used in the smart contracts that could benefit it.

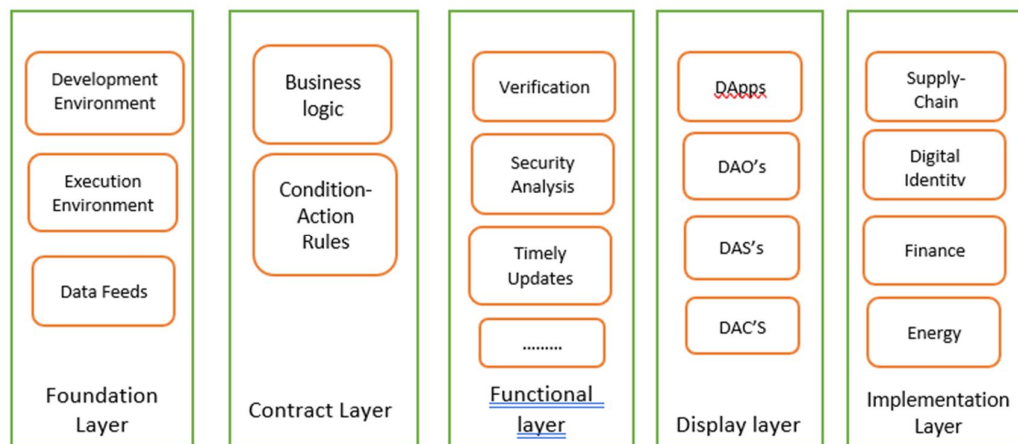


Figure 5: Proposed Framework for Better Modelling of Current Methods involved in Smart contracts

This method can be modelled in a more effective framework that can be divided into a foundation layer, contract layer, functional layer, display layer, and implementation layer—five layers in all.

The foundation layer, shown in the figure starting from the left, contains the working and consists of development environments with a variety of tools like wallets, programming languages, IDEs, etc. with which the smart contracts are developed, deployed, and execution environments that contain the consensus algorithms like proof of work, proof of stake, etc. along with other blockchain key components like the P2P network, cryptography, and incentive mechanism [3].

The contract layer contains information about the contract's terms, regulations, and scenario-based reactions with communication between all parties. The parties involved, the terms of the contract, the legal requirements and guidelines to be followed during execution, the business logic, and other agreements must all be taken into consideration while constructing the smart contracts [9]. These parameters are then given to the programmers who use a variety of methods and tools to translate these components from Basic English into executable code, creating an automated application that can execute all of the given arguments when the necessary conditions are satisfied.

All actions taken to improve and efficiently execute the smart contract are included in the functional layer. These include security analysis and verification, which make sure that the program is running with the correct semantics as intended by the author and that there are no vulnerabilities that could cause harm and eventually have an impact on users or clients. Following the deployment of the smart contracts on the blockchain, regular updates are carried out to guarantee reparability in the extremely unlikely event of record tampering [13].

Display layer encompasses various manifestations of smart contracts, including decentralized applications, autonomous organizations, and societies that will advance traditional business and management by developing into a programmable society that can be easily administered and managed regardless of the processes involved because all the rules will be recorded on the blockchain, lowering transaction costs and aligning the interests of various stakeholders.

The Implementation layer, which can be built on top of the Display layer, completes the stack. For instance [11]. The Decentralized Autonomous Organization (DAO), created in the Ethereum Blockchain, serves as the foundation for the distributed autonomous art application. It creates an organized capital market by bringing together all the artists' audiences and creators in a cooperative connection. Healthcare, digital identity management and verification, finance, and other application domains are also included.

Due to certain limits of technology, tools, and other dependencies that occur while investigating more and more of these decentralized concepts over Blockchain Smart contracts, this is only an ideal framework that has not yet been finalized and deployed in the modern world.

## V. CONCLUSIONS

Blockchain and smart contracts still being very new technologies, store a lot of potential. Emerging smart contracts have become a prominent research issue in both academic and industry groups as a result of the increased popularity and expanded applications of blockchain technology. Smart contracts' decentralization, enforceability, and verifiability properties allow for the execution of contract conditions amongst unreliable parties without the need for a central server or a trusted authority. Thus, it is anticipated that smart contracts would revolutionize a number of established industries, including finance, management, IoT, etc. We provide a thorough review of smart contracts in this paper, covering its advantages and disadvantages as well as possible application situations. We also brought several solutions to the setbacks smart contract faces and ways to encounter them with technologies available.

## REFERENCES

- [1] Bhabendu Kumar Mohanta, Soumyashree S Panda, Debasish Jena "An Overview of Smart Contract and Use cases in Blockchain Technology"
- [2] Hewa, Tharaka; Ylianttila, Mika; Liyanage, Madhusanka" Survey on blockchain based smart contracts: Applications, opportunities and challenges"
- [3] Shuai Wang , Liwei Ouyang, Yong Yuan , Xiaochun Ni, Xuan Han, and Fei-Yue Wang , "Blockchain-Enabled Smart Contracts: Architecture, Applications, and Future Trends"
- [4] Hiroki Watanabe , Shigeru Fujimura , Atsushi Nakadaira , Yasuhiko Miyazaki, Akihito Akuts , Jay Kishigami "Blockchain Contract: Securing a Blockchain Applied to Smart Contract".
- [5] Valentina Gatteschi ,Fabrizio Lamberti , Claudio Demartini , Chiara Pranteda , Víctor Santamaría "Blockchain and Smart Contracts for Insurance: Is the Technology Mature Enough?"
- [6] Konstantinos Christidis, Micheal Devetsikiotis "Blockchains and Smart Contracts for the Internet of Things".
- [7] Zibin Zheng, Shaoan Xie, Hong-Ning Dai, Weili Chen, Xiangping Chen, Jian Weng, Muhammad Imran, " An Overview on Smart Contracts: Challenges, Advances and Platforms".
- [8] Lauslahti, Kristian; Mattila, Juri; Seppälä, Timo (2017) : "Smart Contracts – How will Blockchain Technology Affect Contractual Practices?", ETLA Report, No. 68, The Research Institute of the Finnish Economy (ETLA), Helsinki
- [9] Blockchain Research Lab, Colonnaden 72, 22303 Hamburg , University of Hamburg, Faculty of Business, Economics & Social Sciences, "Smart Contracts on the Blockchain – A Bibliometric Analysis and Review"
- [10] Reggie O'Shields" Smart Contracts: Legal Agreements for the Blockchain".

- [11] Zaleha Fauziah ,Haznah Latifah , Xavier Omar ,Alfiah Khoirunisa ,Shofiyul Millah , “Application of Blockchain Technology in Smart Contracts: A Systematic Literature Review”
- [12] Maher Alharby, Aad van Moorse, “Blockchain based smart contracts: a systematic mapping study” .
- [13] Staples, M., Chen, S., Falamaki, S., Ponomarev, A., Rimba, P., Tran, A. B., Weber, I., Xu, X., Zhu, J., (2017) Risks and opportunities for systems using blockchain and smart contracts. Data61 (CSIRO), Sydney.
- [14] Adam P. Balcerzak , Elvira Nica , Elzbieta Rogalska , Miloš Poliak , Tomáš Klieščík ,Oana-Matilda Sabie” Blockchain Technology and Smart Contracts in Decentralized Governance Systems”.
- [15] Tharaka mawanane hewa, yining hu, madhusanka Liyanage, salil s kanhare and mika ylianttila, ,” Survey on Blockchain-Based Smart Contracts: Technical Aspects and Future Research”.
- [16] T. Chen, X. Li, X. Luo, and X. Zhang, “Under-optimized smart contracts devour your money,” in 2017 IEEE 24th International Conference on Software Analysis, Evolution and Reengineering (SANER), pp. 442-446, IEEE, 2017.
- [17] F. Idelberger, G. Governatori, R. Riveret, and G. Sartor, “Evaluation of logic-based smart contracts for blockchain systems,” in International Symposium on Rules and Rule Markup Languages for the Semantic Web, 167-183, Springer, 2016.
- [18] C. Natoli and V. Gramoli, “The blockchain anomaly,” in 15th International Symposium on Network Computing and Applications (NCA), 310-317, IEEE, 2016.
- [19] L. Luu, D.-H. Chu, H. Olickel, P. Saxena, and A. Hobor, “Making smart contracts smarter,” in Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, CCS '16, pp. 254-269, ACM, 2016.