International Journal of Enhanced Research in Science, Technology & Engineering
ISSN: 2319-7463, Vol. 11 Issue 12, December-2022, Impact Factor: 7.957

A Review on Hyperledger fabric for business model

Girija Rani Suthoju[1], M. Suresh Babu[2]

[1]Dept. of CSE, Neil Gogte Institute of Technology, Hyderabad, TS
[2]Dept. of CSE, YV Sivareddy College of Engineering, Anantapur, AP

ABSTRACT

Hyperledger Fabric is an enterprise-grade permissioned distributed ledger framework. Hyperledger Fabric differentiates itself from other Blockchain system with private and permissioned network. Developed by Linux Foundation, any entity can enroll through its trusted Membership Service Provider (MSP). Hyperledger Fabric is a modular blockchain framework that acts as a foundation for developing blockchain based products, solutions, and applications using plug-and-play components that are aimed for use within private enterprises. Our focus is on developing a modular architectural framework for enterprise-class distributed ledgers. This includes identifying common and critical components, providing a functional decomposition of an enterprise blockchain stack into component layers and modules, standardizing interfaces between components, and ensuring interoperability between ledgers.

Key terms: Hyperledger, block chain, chain nodes, transaction, endorsement, commitment, network.

INTRODUCTION

Hyperledger Fabric is a modular blockchain framework that acts as a foundation for developing blockchain-based products, solutions, and applications using plug-and-play components that are aimed for use within private enterprises. Our focus is on developing a modular architectural framework for enterprise-class distributed ledgers. This includes identifying common and critical components, providing a functional decomposition of an enterprise blockchain stack into component layers and modules, standardizing interfaces between components, and ensuring interoperability between ledgers.

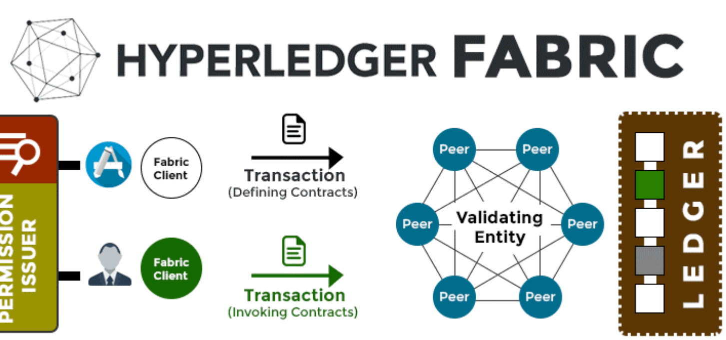

Figure: 1

LITERATURE SURVEY

The introduction of blockchain technology came as a result of mistrust in the traditional model of transaction handling, as stated by Satoshi Nakamoto the inventor(s) of Bitcoin; "Commerce on the internet has come to rely almost exclusively on financial institutions serving as trusted third parties to process electronic payments" [2]. Bitcoin initiated the rise of popularity surrounding the blockchain technology by introducing a decentralized, distributed, and transparent ledger technology that can maintain itself. Although not without faults, the technology is promising. In the subsequent years, interests in blockchain technology grew, with more and more institutions trying to find possible uses for the technology outside of crypto currencies. Blockchain technologies are now not limited to Financial technologies. [6]. Many of the areas mentioned that want to incorporate blockchain have specific requirements that public blockchains are not able to meet, specifically the transparency and the membership control. In cases where privacy in some form and the control of access to the network are required, simple public blockchains are not sufficient. From their research to find solutions to these problems, IBM has managed with the cooperation with the Linux Foundation, to start development of a permissioned based blockchain [7]. These types of blockchain are becoming more popular with enterprises as they provide membership control, privacy, and a central governing entity. The properties that were so popular with public blockchains are still in use, permissioned blockchains are decentralized and distributed, but with the addition of a network administrator that can manage membership control and access control. Decentralization in permissioned blockchains has different properties than in public blockchains, which do not have a governing entity, and all processes are done peer-to-peer. In permissioned blockchains a central governing entity is present, in order to maintain membership control and ledger control, the processing and handling of data is done similarly to the public blockchains, and can, as a result, be described as decentralized. 3 These properties will inturn reinstate a form of trust between the network members, compared to the non-trust environment of public blockchains. The decentralized and distributed manner of blockchain would strengthen network resiliency and data storage, at the same time, the permissioned nature of a permissioned blockchain would enable membership control and limit the attack surface. The capabilities of the blockchain network enable the data stored to be auditable and trustworthy due to the hashing processes undergone in the network. This thesis specifically focuses on the implementation of a permissioned blockchain with the use of the Hyperledger Fabric framework, to fulfill the requirements of privacy, security and network availability, the CIA-triad will be set as a requirement that the blockchain .

HYPERLEDGER FABRIC WORKING ARCHITECTURE

Traditional blockchain networks can't support private transactions and confidential contracts that are of utmost importance for businesses. Hyperledger Fabric was designed in response to this as a modular, scalable and secure foundation for offering industrial blockchain solutions.

Hyperledger Fabric is the open-source engine for blockchain and takes care of the most important features for evaluating and using blockchain for business use cases.

Within private industrial networks, the verifiable identity of a participant is a primary requirement. Hyperledger Fabric supports memberships based on permission; all network participants must have known identities. Many business sectors, such as healthcare and finance, are bound by data protection regulations that mandate maintaining data about the various participants and their respective access to various data points. Fabric supports such permission-based membership.

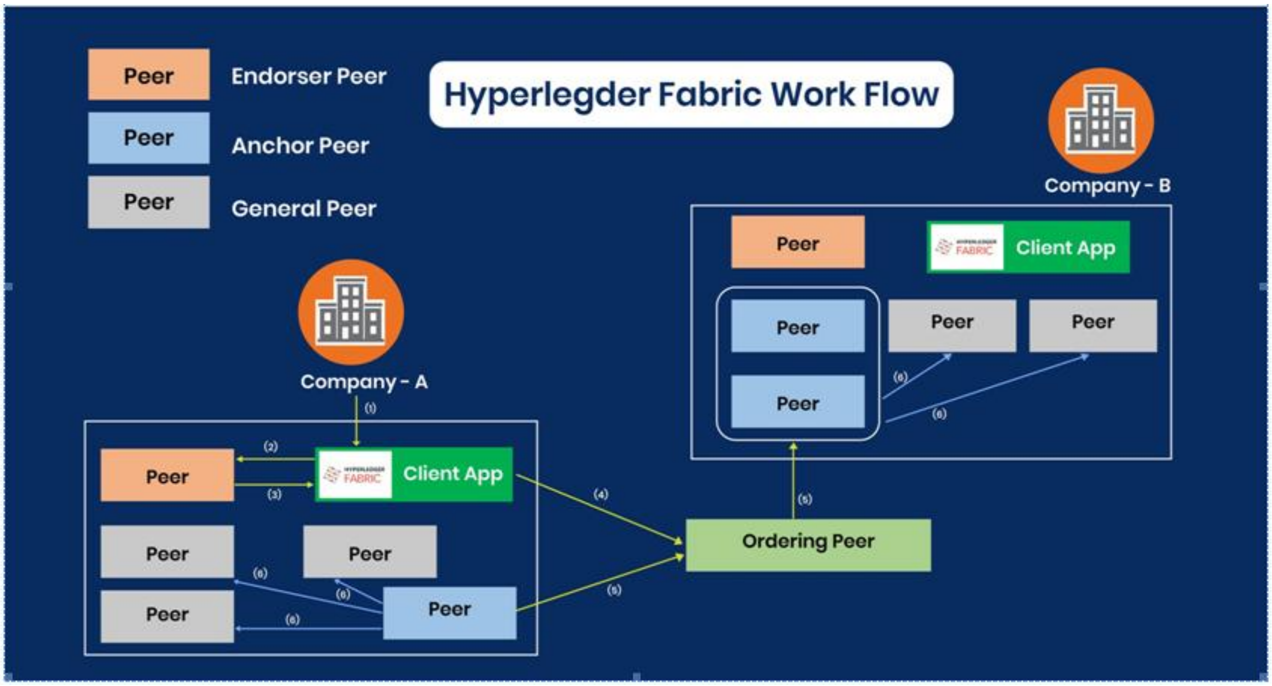

Figure: 2

The modular architecture of Hyperledger Fabric separates the transaction processing workflow into three different stages: smart contracts called chain code that comprise the distributed logic processing and agreement of the system, transaction ordering, and transaction validation and commitment. This segregation offers multiple benefits:

- A reduced number of trust levels and verification that keeps the network and processing clutter-free
- Improved network scalability
- Better overall performance

Additionally, Hyperledger Fabric's support for plug-and-play of various components allows for easy reuse of existing features and ready-made integration of various modules. For instance, if a function already exists that verifies the participant's identity, an enterprise-level network simply needs to plug and reuse this existing module instead of building the same function from scratch.

The participants on the network have three distinct roles:

- Endorser
- Committer
- Consenter

In a nutshell, the transaction proposal is submitted to the endorser peer according to the predefined endorsement policy about the number of endorsers required. After sufficient endorsements by the endorser(s), a batch or block of transactions is delivered to the committer(s). Committers validate that the endorsement policy was followed and that there are no conflicting transactions. Once both the checks are made, the transactions are committed to the ledger.

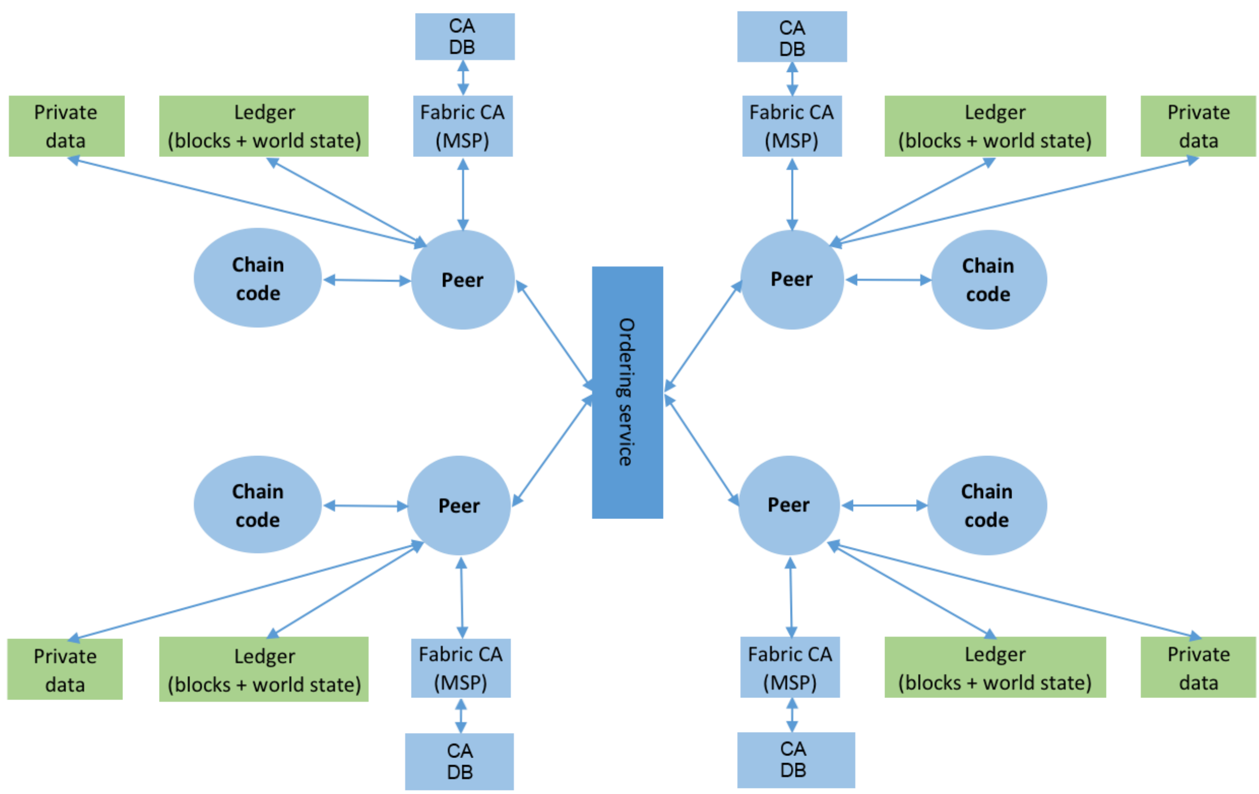

Figure: 3

SMART CONTRACTS IN HYPERLEDGER FABRIC

A smart contract in Hyperledger Fabric is a program, called chaincode. Chaincode can be written in Go, JavaScript (node.js), and eventually other programming languages such as Java that implement a prescribed interface. Chaincode runs in a secured Docker container isolated from the endorsing peer process. Chaincode initializes and manages the ledger state through transactions submitted by applications. A chaincode typically handles business logic that members of the network have agreed to. The state created by a chaincode is scoped exclusively to that chaincode and can't be accessed directly by another chaincode. However, with the appropriate permission, a chaincode in the same network can invoke another chaincode to access its state. There are two different types of chaincode to consider: • System chaincode • Application chaincode System chaincode typically handles system-related transactions such as lifecycle management and policy configuration. However the system chaincode API is open for users to implement their application needs as well. Application chaincode manages application states on the ledger, including digital assets or arbitrary data records. A chaincode starts with a package that encapsulates critical metadata about the chaincode, including the name, version, and counterparty signatures to ensure the integrity of the code and metadata. The chaincode package is then installed on the network nodes of the counterparties. An appropriate member of the network (as controlled by policy configuration) activates the chaincode by submitting an instantiation transaction to the network. If the transaction is approved, the chaincode enters an active state where it can receive transactions from users via client-side applications. Any chaincode transactions that are validated are appended to the shared ledger. These transactions can then modify the world state accordingly. Any time after a chaincode has been instantiated, it can be upgraded through an upgrade transaction. Using Chaincodes to Develop Business Contracts and Decentralized Applications There are generally two ways to develop business contracts for Hyperledger Fabric:

- To code individual contracts into standalone instances of chaincode.
- To use one chaincode to handle all contracts (of certain types) and have it expose APIs to manage lifecycle of those contracts. This second approach is probably more efficient. Using Chaincodes to Define and Manage Assets Users of Hyperledger Fabric can also use chaincode to define assets and the logic that manages them. 11 In most blockchain solutions, there are two popular approaches to defining assets:
- The stateless UTXO (unspent transaction output) model, where account balances are encoded into past transaction records.
- The account model, where account balances are kept in state storage space on the ledger Each approach has benefits and drawbacks. Hyperledger Fabric does not require one over the other. Instead, it ensures that both approaches are easy to implement.

LIMITATIONS

We limit the focus of this thesis to a permissioned blockchain based on the Hyperledger Fabric framework, specifically version 1.2. The variety of permissioned blockchains has grown over the years, but due to time and resource constraints, we cannot test all to find the most suitable. The environment of the project and application are limited due to our resource and time constraints. We have tried to create an environment that is as realistic as we can manage. The performance tests are based on the Hyperledger Caliper testing framework developed to support benchmarking and performance testing of Hyperledger frameworks; this causes the metrics of the performances to be limited to the provided metrics of Caliper. Due to Hyperledger Fabric being relatively new and still under development, issues arose throughout the lifecycle of this project. The documentation of the framework is quite complicated and continuously under construction, requiring some time to understand the development process and implementation. The lack of documentation on a remote deployment of a Hyperledger fabric blockchain and the abstract error messages were quite demanding and time-consuming. Additional tools provided by Hyperledger Fabric that we used early in the project for development were later deprecated, causing us to re-implement the whole project

CONCLUSION

This paper explores the modular architecture used by all Hyperledger projects and looked at the different ways smart contracts can be implemented within this modular framework.

REFERENCES

[1]. https://techcrunch.com/2020/01/30/hyperledger-fabric-the-open-source-distributed-ledger-reaches-release-2-0/ IBM. "What Is Hyperledger Fabric?"
[2]. Satoshi Nakamoto. 'Bitcoin: A peer-to-peer electronic cash system'. In: (2008). URL: https://bitcoin.org/bitcoin.pdf

[3]. Hyperledger umbrella project. Hyperledger Caliper Documentation site. 2018. URL: https:// hyperledger.github. io/ caliper/ docs/1_Getting_ Started.html (visited on 24/06/2019).
[4]. Hyperledger and the Linux Foundation. Hyperledger fabric whitepaper. 2018. URL: https://www.hyperledger.org/wp - content/uploads/2017/ 08 / Hyperledger _ Arch _ WG _ Paper _ 1 _ Consensus . pdf (visited on 31/05/2019)
[5]. Docker Inc. Docker glossary: Docker. 2019. URL: https://docs.docker. com/glossary/?term=Docker (visited on 31/05/2019)
[6]. Zibin Zheng et al. 'Blockchain challenges and opportunities: A survey'. In: International Journal of Web and Grid Services 14.4 (2018), pp. 352–375
[7]. IBM. Hyperledger Blockchain collaboration. URL: https://www.ericsson. com / en / mobility - report / internet - of - things - forecast (visited on 25/07/2019).
[8]. R. Roman, P. Najera and J. Lopez. 'Securing the Internet of Things'. In: Computer 44.9 (Sept. 2011), pp. 51–58. ISSN: 0018-9162. DOI: 10 . 1109/MC.2011.291.

AUTHORS

[1] Ms. GIRIJA RANI SUTHOJU pursuing Ph.D in the Department of Computer Science and Engineering at JNTU Hyderabad. She received Masters degree (M.Tech) in the Department of CSE at Aurora's Technological and Research Institute belongs to JNTU, Hyderabad in 2014 and Bachelors degree (B.Tech) in the Department of Computer Science and Engineering from the same University in 2012. She worked as an Assistant Professor for 5years in Aurora's Technological and Research Institute affiliated to Jawaharlal Nehru Technological University, Hyderabad. She worked in KLEF (K L deemed to be University) Hyderabad from June 2019 to June 2022 and at present she is working in NGIT Hyderabad. She also received Teaching Excellence award for best teaching thrice from 2016 to 2018 from Aurora consortium and also in 2021 by KL University for her best performance in academics. She also received Global Eminent Teacher Award recently. Her area of research is Cryptography and network security.

[2] M. SURESH BABU completed PhD (Computer Science) from Sri Krishnadevaraya University, Anantapur, M.Phil (Computer Science) from Bharthiar University in 2007, Master of Computer Applications from Osmania University,Hyderabad in 1997, PGDBA from Sri Krishnadevaraya University, Anantapur in 2002, DCOM from IGNOU, New Delhi and Bachelor of Science from Sri Krishnadevaraya University, Anantapur in 1993. His area of interests are Datamining, Cloud computing and Networks. He is a creative professional with around 18+ years of experience in Teaching & Student Management. Presently, he is working as Principal, in Y V Sivareddy college of Engineering, Ananthapur, AP. He worked as a Professor in Department of Computer Science & Engineering, S K D Engineering College, Ananthapur, AP and also as a Principal, Aurora's Technological Institute, Uppal, Hyderabad from Jan- 2015 Aug-2015, also as Principal, Intel Institute of Science, Anantapur,from October 1998 - December 2012. He worked as Professor & Head, Department of Computer Applications, Madanapalle Institute of Technology & Science, from December 2012 to December 2014.