Decentralized and Automated Business Logic in a Future MBSE Blockchain Ecosystem

Red Boumghar*, Annalisa Riccardi†, Shahzad Ameen‡, Ashwin Arulselvan§, Edmondo Minisci¶

Blockchain represents a ready-to-use framework to distribute information, record changes and agree, by consensus, upon the global state of represented things. Blockchain systems can be equipped with a layer that can model the business logic in a pre-programmed and automated way. This is achieved through the use of smart contracts, that are computer programs stored on a blockchain and visible to all participants. Smart contracts represent digital assets that can be triggered upon met pre-conditions. They can be used to represent the various items of the MBSE domain such as requirements, system engineers, missions, spacecrafts and subsystems. Creating digital twins of these components along with their representation as assets in the blockchain enables multitude of use cases for flexible automation and integration in various workflows. This papers illustrate an innovation prototype in implementation; Exochain, a blockchain approach to MBSE.

I. Introduction

During early design phase of a mission, the needs and requirements have to be communicated to various domain specific teams. Thus there is a very early demand in sharing items such as documents, models and information across stakeholders before any preliminary design. These items represents digital engineering assets, as mentioned in [1]. Sharing these assets must maintain a coherent understanding of the project amongst stakeholders, and preserve a high level of interoperability. This is still a challenging task in an ecosystem where stakeholders use different design tools, different file formats and heterogeneous and centralized storage and processing facilities.

The acceleration of digitalization in the MBSE domain requires a more open and reliable data and information exchange layer. Several initiatives already contributes to this digitalization (e.g. MBSE/Data Hub) and already try to unify all exchanges of information in a centralized system. This rises the risk of single point of failure. Additional this does not solve trust issues that may arise in sharing digital assets and during stakeholders modifications assessments.

NASA's suggested implementation plans for MBSE [2] mentions an increasing presence of automation and intelligence around mission conception and maintenance. Blockchain technology offers a layer where coherent data distribution, changes monitoring, full traceability and non repudiation (thanks to cryptography) are already available. Blockchain can thus be used as a layer for business automation by encoding system engineering protocols and services in smart contracts. Additionally it can be used as a market place, a zone of exchange of digital engineering assets.

The presented project proposes a blockchain platform, named *Exochain*, which sets up the necessary business mechanics to engage in automation supported concurrent design.

II. Blockchain Ecosystem

A. Digital Artifacts and automatic generation

In order to represent any digital assets on the blockchain, open source semantic definitions and ontologies [3, 4] are used to shape their corresponding metadata. Challenge is that many digital assets are domain specific and a generic enough approach needs to be deployed. The following Ethereum proposal standards are used:

- ERC20: A standard interface for tokens. It provides basic functionality to transfer tokens, as well as allow tokens to be approved so they can be spent or received by another on-chain third party.

*Parametry.ai, red@parametry.ai
†University of Strathclyde, annalisa.riccardi@strath.ac.uk
‡Parametry.ai, shahzad@parametry.ai
§University of Strathclyde, ashwin.arulselvan@strath.ac.uk
¶University of Strathclyde, edmondo.minisci@strath.ac.uk

- ERC725: A standard interface for a smart contract based account that can be used by humans, groups, organisations, objects and machines with attachable key value store.
- ERC1155: A standard interface for single deployed contract that may include any combination of fungible tokens, non-fungible tokens or other configurations (e.g. semi-fungible tokens).

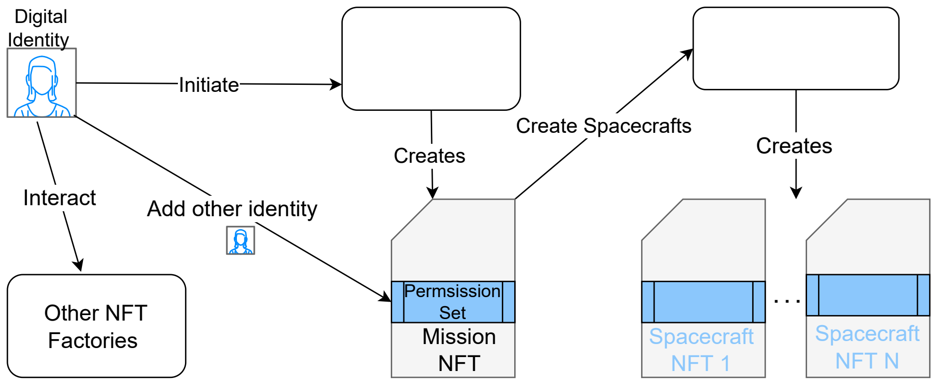

Fig. 1 Exochain digital assets factories. Factories are smart contracts to generate other smart contracts; e.g. Non-Fungible Tokens.

B. Difference with Version Control Systems

Certain traditional version control systems such as **git** are distributed by design. Their level of automation depend on additional software solutions (github or gitlab) which provides continuous integration and deployment capabilities. Amid business logic automation, blockchain offers a distributed coherent state maintenance that cannot be found in traditional systems.

- Traditional version control (TVC) has one repository while exochain is capable of holding the assets of several missions with interoperability across missions and assets
- TVC employs generic integration scripts for automation while exochain employs MBSE oriented smart contracts
- TVC employs a central repository which are prone to merge conflicts. Concurrent edits are avoids conflict by design as only "permanently written" transactions will be recognised as authentic in the distributed repository of exochain. Subsequent transaction will be signed off only if they are compatible with the previous transactions.
- Changes in TVC are verified by peers manually, while verification in exochain can be automated
- Permission management in TVC is centralised while its collaborative in exochain
- Project management is collaborative in both TVC and exochain. Scripts and smart contracts could enable built-in business automation
- Journaling system is unified in exochain as against the multiple sources employed in TVC

C. Difference with current MBSE practice

Workshops were conducted alongside interviews with users of MBSE systems during which the participants shared their user experience. Some of the current issues highlighted can be divided into five main groups: accountability, interoperability, trust, governance and human factors. The main key points and current challenges have been listed here below

- in terms of transparency and accountability: need of ability to do forensics and tracking of accesses and changes. This is intrinsically designed within Exochain or any other blockchain system. Transactions are recorded unequivocally into an immutable ledger including record of changes and authoring. No duplication, unauthorised and un-authored changes are allowed by design within a blockchain system,
- in terms of interoperability: a common standard or vocabulary that is capable of checking compatibility and integrity is not often present. The development of the first Space System Ontology, by the OSMoSE working group is addressing this issue and will be pivotal for the development and deployment of smart contracts within Exochain,
- in terms of trust and governance: it has been identified the need to trust new service providers on aspects where technical visibility is more limited than usual. The blockchain infrastructure by design can support zero knowledge

proof protocols that can help addressing this issue. Exochain can be deployed also as a global configuration layer for MBSE,
- in terms of human factors: We need to define ways to include human aspects (team management, etc.) into the MBSE model, along with its technical aspects. The human aspect in Exochain is decoded trough the development of digital identities and voting mechanisms.

Although blockchain provides the tools for trust management, deploying blockchain systems comes with its own challenges: from the technical ones of developing more complex and secure decentralised systems to the changes in human dynamics where the network of interactions is closer to the one of a peer to peer network rather than a centralised pyramidal one.

III. Current implementation

A. Overall Architecture

The general system architecture is shown in figure 2. This architecture has an application layer on the backend composed of services that provide blockchain activity monitoring capabilities. The blockchain itself (ensemble of nodes participating in the network) is a private collection of nodes ran by selected participants. The machines participating in this blockchain as nodes would reside within the EU boundaries and would fulfill GDPR regulations. There will be no link with public blockchains, therefore no possible information leaks to external third parties assuming the network is correctly set up and access strictly reserved to participating entities and personnel.

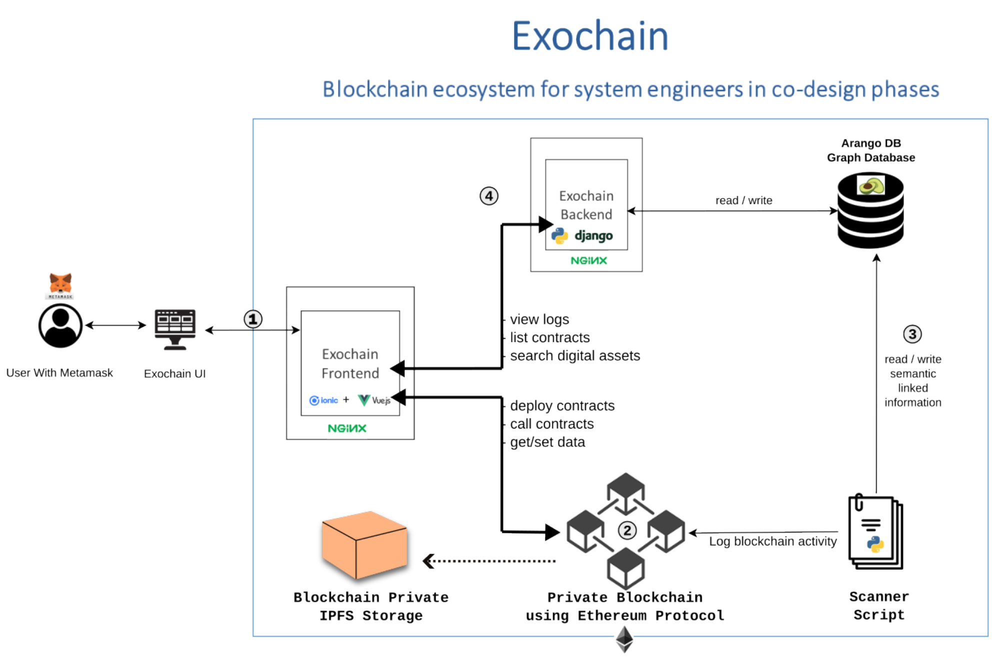

Fig. 2 Exochain blockchain architecture with applied technologies

The components in figure 2 are the following:
- **Step 1**: On the left, **users** access a decentralized application web page. This web page is served as any other classic pages on the internet via a **frontend server**. The difference is that user has to connect with their wallet applications, the decentralized application does not hold any user credentials. Once connected, the user can deploy and call contracts.
- **Step 2**: From the frontend, certain actions are directly sent to the blockchain which is an **Ethereum compatible private blockchain** with a JSON-RPC (Remote Procedure Call) interface. Actions are, for example, deploying contracts or calling digital asset factories (smart contracts to build digital assets). Metadata for each digital assets is stored on a blockchain dedicated distributed storage using the IPFS filesystem [5]. Metadata content can be

encrypted in order to fulfil non-public accessible information needs.

- **Step 3**: In order to list all profiles accessible by one address we have to retrieve and parse all information in all blocks every time a user requests information. This process is not scalable as the blockchain grows. To prevent this we use a **scanner python script** that explores the blockchain and listens to new event and then saves them and tags them in a graph database as they come. The **graph database** offers an administration interface to check what information was saved there and manually draw graph queries to see the links between users, their digital identities and MBSE digital assets represented on the blockchain.
- **Step 4**: The information that is semantically indexed in this database is then fastly served by our **backend API** to the user interface in order to get information much faster than if having to scan the whole blockchain for every request.

B. Exochain interfaces to industry MBSE software

The Exochain web front end permits anyone with a standard browser to complete all available actions on Exochain.

The web-based interface can be replaced by a connector on a dedicated MBSE software. It would report activities of the MBSE software user by submitting change and updates on models to the Exochain blockchain. Nevertheless this requires efforts proportional to the number of different software in the industry.

Standard file formats are also a viable interface, whether they are directly provided by a concurrent design software or by a data hub which already implements the different translations to this standard file format. The interfaces are capable of handling security privileges that imposes varying restrictions to data and assets depending on the users' accessible identities.

C. Engineering Asset Privacy

Identity management layer over the blockchain provides security restrictions to access the repository. In addition, assets can be stored in encrypted format in the repository that allows a further layer of security. Cryptographic primitives native to blockchain enables zero-knowledge protocols to be efficiently implemented permitting approval and acceptance of proof of concepts without revealing sensitive intellectual property information, neither being able to reverse engineer them. This allows collaboration to happen over confidential information.

IV. Example Scenario

The bootstrapping scenario shows how concurrent design of a mission could be initialized on a blockchain such as Exochain. This set of steps shows how digital assets factories are used and how permissions are set to control who can do what on this distributed system.

1) Create your digital identity D_1 (more complex set-ups are possible and not explanained here)
2) Call the mission factory contract to initiate a mission with D_1; a mission is a Non Fungilble Token, NFT
3) Create requirements for this mission and the requirements will conform to a common set of vocabulary provided by an open source vocabulary defined in projects such as Metasat [4]. Digital identity D_1 a Mission NFT M_1 created using defined requirements. D_1 can provide varying degree of access to multiple other identities.
4) One or multiple spacecraft NFTs ($SC_1..SC_N$) are created via the spacecraft factory contract by one of the identities having the required access to M_1.
5) Subsystem ($SS_1..SS_M$) are created for each spacecraft or for a general catalog via a subsystem factory. This results in a knowledge graphs between missions, spacecraft and its subsystems. This graph uses defined common vocabulary set. The creation of Subsystem NFT is independent and can be done by any claimed digital identity.
6) If the creator D_n wants the created Subsystem NFT SS_1 included in the spacecraft NFT (for instance SC_1) then she/he would send a request to the Spacecraft NFT SC_1's owner (D_x) for permission to include this Subsystem in the spacecraft, generally to fulfil a design proposal.
7) Based on the requirements of spacecraft and mission, the identity can include or reject the subsystem in the mission/spacecraft and therefore resulting in changes on meta information of the Spacecraft NFT.

Each kind of NFT can be equipped with various set of business functions. They represent business logics, like for instance creating a vote proposal upon a change request to the asset. Auto-tagging changes or broadcasting information updates to a specific group of people under specific conditions that can be automatically checked are such functions that can be implemented.

V. Conclusion

Using a blockchain approach to manage mission design and conception allows freedom of implementation of simple to complex business logics. Thanks to bridges with the external world, interactions with external services such as procurement are becoming possible.

This opens to a ensemble of new automation possibilities where trust is managed by design thanks to the consensus mechanism of the underlying blockchain protocol (for our prototype; Ethereum's Proof of Stake protocol). The shift of paradigm of this approach can push the concurrent design ecosystem to become a decentralized autonomous organization (DAO); all stakeholders can participate in the ecosystem governance, such as task distribution and deliverable verification. Incentives are created for stakeholders to share information and decisions are taken by the network, by all stakeholders. Procurement activities can use what happened on the blockchain, with the support of a *proof of deliverable*, to also automate activation of real world contracts (payments, restricted information accesses, etc.).

Identified first steps in this direction shall implement automatic generation of smart objects (NFTs) in a single source of truth (a hub). Then business logic can be implemented in more details with richer external services interactions.

We finally conclude with a note on some of the challenges in Exochain. Although interoperability across missions and associated assets is possible within Exochain, interoperability with missions set up on other blockchains set with the same standards is still a challenge. Once a mission with identities and smart contracts to enable automation have been set up, Exochain is easy to adopt but the initial set up would require a significant amount of skill and knowledge on blockchain.

References

[1] Lu, J., Zheng, X., Hu, Z., Zhang, H., and Kiritsis, D., "Towards a Decentralized Digital Engineering Assets Marketplace: Empowered by Model-based Systems Engineering and Distributed Ledger Technology," , May 2020. doi:10.48550/arXiv.2005.05415, URL `http://arxiv.org/abs/2005.05415`, number: arXiv:2005.05415 [cs, eess].

[2] Knizhnik, J., Weiland, K., Grondin, T., and Holladay, J., "Suggested MBSE Implementation Plan Approaches," , ???? URL `https://ntrs.nasa.gov/citations/20205006908`.

[3] Hennig, C., Viehl, A., Kämpgen, B., and Eisenmann, H., "Ontology-Based Design of Space Systems," *The Semantic Web – ISWC 2016*, edited by P. Groth, E. Simperl, A. Gray, M. Sabou, M. Krötzsch, F. Lecue, F. Flöck, and Y. Gil, Springer International Publishing, Cham, 2016, pp. 308–324. doi:10.1007/978-3-319-46547-0_29.

[4] John G. Wolbach Library, L. S. F., "Open, collaboratively-developed metadata to support the future of space exploration," , 2022. URL `http://metasat.schema.space/`.

[5] Juan Benet, e. a., "Academic papers | IPFS Docs," , 2022. URL `https://docs.ipfs.io/concepts/further-reading/academic-papers/#ipfs-content-addressed-versioned-p2p-file-system`.