# A Blockchain Based Decentralized Computing And NFT Infrastructure For Game Networks

Koushik Bhargav Muthe, Khushboo Sharma, Karthik Epperla Nagendra Sri
*CSE Department*
*SRM University AP*
Amaravati, India
Email: koushik_bhargav@srmap.edu.in, khushboo_sharma@srmap.edu.in, epperla_nagendra@srmap.edu.in

***Abstract*—The market value of the Gaming industry was said to be over 138 Billion USD in 2019. Competitive Gaming or eSports was already included in the Asian Games 2020, and the Olympic committee is considering including eSports into Olympics 2024. The online gaming segment amounts up to 7 quintillion bytes of internet traffic every month. Most of this data is controlled by centralized gatekeepers like cloud agencies and game creators. This is causing many issues ranging from privacy concerns to latency. The game makers have complete rights over the game to arbitrarily change the rules, set prices for the assets, and control over game servers. Even the reward mechanisms in most of the games, including eSports, are controlled by the game producers, and these rewards have no real-world value. The motivation behind this research is to create a decentralized computation and token management infrastructure for game networks. This paper focuses on using Ethereum Blockchain, IPFS and ERC - Ethereum Request for Comment 1155 architecture to build a gaming-oriented public decentralized network.**

***Index Terms*—Cloud Gaming, Ethereum Blockchain, Decentralized Computation, IPFS - Inter Planetary File System, ERC 1155 - Ethereum Request for Comment.**

## I. Introduction

Gaming is one of the largest spaces in the Entertainment Industry. It is massive, with annual revenue of over 138 Billion USD in 2019 [1]. It also has an enormous audience, with over 2.47 Billion gamers around the world by 2019 [2]. Competitive Gaming or eSports is another primary market in Gaming with projected revenue of 2.96 Billion USD by the end of 2022 [3]. It was included in the Asian Games 2018 as a demonstrational sport and the Olympic committee has considered to include it in the 2020 Summer Olympics in Tokyo. Provided with these rapid changes in the gaming industry game makers are spending billions of dollars into building and improving the games in the market. The increase in the number of people with access to the internet has increased the amount of online Gaming which accounts for seven quintillion bytes of internet traffic every month. Very few significant entities control the gaming industry. Even the large volume of internet traffic is controlled by some large corporations who act as gatekeepers for the gaming network. Gaming is no more an act of leisure as the scope of the gaming industry has increased a lot in the past few years. This applies to the contribution made by players across various gaming platforms which includes their profiles and assets. Gamers spend a lot of money in the form of fiat for purchasing in-game assets, and these assets have no real-world value. The game producers control the prices of the assets, and the gaming community has no stake in it. There are also no storage spaces and real-world market places that can support secure storage and trading of these in-game assets. Gamers often face latency issues because of the centralized gatekeeper architecture of the gaming network [4]. The gaming community should have more stake in the decisions made by the game creators extending from prices of in-game assets to game features.

The primary focus of this paper is to propose a gaming network where the game players are given a stake in the gaming industry. A decentralized computing architecture would provide a solution to this by eliminating centralized game servers. The Ethereum blockchain would provide gamers a real-world value for their intangible assets by converting them into NFTs - Non Fungible Assets, which are rare and unique blockchain managed digital assets. These assets have a real-world value and can be traded in markets outside the game network. IPFS is a peer to peer hypermedia protocol which is a reliable Web3.0 based protocol. IPFS removes the need for centralized entities such as game servers and connects the nodes in the network directly. This can potentially reduce the latency and also provide a secure channel for gaming. Ethereum 2.0 aims to introduce proof of stake architecture into the leading network, enabling gamers to have a stake in the network, which can be utilized to improve their game features. This can reduce the complete control of games by game makers and also helps game developers to add better features which are in consensus with the gaming community.

## II. Background and Related Works

Cloud gaming is aimed to be one of the most revolutionizing technologies in the gaming industry. However, the first few glimpses from major players such as Google and Microsoft were not playable. Issues such as latency and low quality are consistently seen across all services. The work by Mark Claypool and David Finkel clearly stated the effects of latency on the performance of the players [5]. The work done by Bryce Mariano and Simon G. M. Koo clearly stated that Cloud Gaming is nearly impossible if it depends on the current internet architecture [6]. The paper by Daniel Uribe and Gisele

978-1-7281-8370-1/20/$31.00 ©2020 IEEE


Authorized licensed use limited to: Univ of Calif Santa Barbara. Downloaded on May 17,2021 at 18:05:47 UTC from IEEE Xplore. Restrictions apply.

Waters presented the privacy and decentralized advantages of using NFTs but is limited to Genome Research [7]. The limitations of the previous works include dependency on centralized server-based communication for gaming and limited inclusion of Blockchain-based NFTs for game objects or assets. The proposed protocol has an advantage over the previous works as it utilizes IPFS and Proxy Computation, which eliminates the need for a centralized game server. This considerably reduces the latency issues. It also integrates Ethereum ERC-1155 for NFT based game objects to create and distribute game objects in a decentralized infrastructure.

### A. Ethereum Blockchain

A Blockchain is an immutable distributed ledger with multiple blocks enchained together, and every block stores transactions in such a way that it's unacceptable to change these transactions. It's an enormous step forward in terms of decentralized and distributed applications. Blockchain technology guarantees advantages in trustability, collaboration, organization, identification, quality, and transparency. Decentralization in easy terms implies that the service or application is deployed on a network in such a way that it provides no comprehensive management over data and execution to any server. No one inside this cluster can vary or delete the previous transactions; instead, every server includes a current copy of data and execution logic. Distributed implies that any server or node on a network is connected to each alternative node directly or indirectly. Ledger is associated with an accounting term, and you can consider it as specialized storage and retrieval of data.


Fig. 1. Blockchain - Block Diagram

Ethereum is the implementation of Blockchain and permits extending its practicality with the assistance of smart contracts. Vitalik Buterin first planned the kernel of this work in Nov 2013 [8]. The state in Ethereum denotes the balances of the account and potential additional data. The main aim of Ethereum is to validate transactions from statements, update their status (state), and continue to maintain that state as the current state until another transaction is approved. One key goal is to facilitate transactions between willing people who would otherwise do not have any means to trust each other. This could be due to geographical separation, interfacing problems, or maybe the incompatibility, censorship, disposition, privacy, expense, uncertainty, or inconvenience.

### B. Smart Contracts

A Smart Contract is a digital set of rules and regulations that expedite, verify, or implement the performance of a contract. Smart contracts are used to validate the credibility of transactions without any third party. Nick Szabo is the pioneer in developing the concept of Smart Contracts [9]. Smart Contacts made Ethereum robust and scalable to multiple dimensions. Solidity and Vyper are two prominent languages to develop Smart Contracts.

### C. EVM - Ethereum Virtual Machine

Ether is the cryptocurrency used in the Ethereum blockchain. Any transaction between accounts is expressed as a distributed currency by Ether. The Ether is the fuel for operation in Ethereum. This permits many applications, starting from the exchange of cryptocurrencies to financial applications, storing and managing tokens and digital assets, conventional systems, identity management, and ballot systems, up to those applications that need traceability resources and assets. EVM - Ethereum Virtual Machine, is the runtime surroundings for smart contracts in Ethereum [10]. It is not solely sandboxed; however, really fully isolated, which suggests that code running within the EVM has no access to the network, filesystem, or alternative processes. Smart contracts even have restricted access to alternative smart contracts. Ethereum Virtual Machine guarantees security by preventing Denial-of-service attacks, that are somewhat associated with rising challenges within the crypto business. Secondly, EVM interprets and executes Ethereum programming language and make sure that communication may be achieved with no interference.

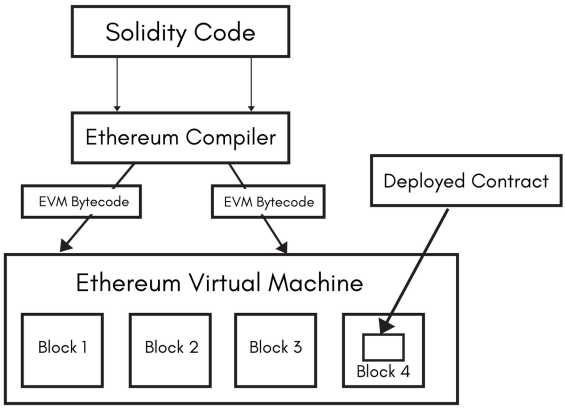


Fig. 2. Ethereum Virtual Machine

### D. IPFS - Inter Planetary File System

Blockchain is a secure medium for data storage, but because of its computational and network limitations, it is not ideal for handling large amounts of data. For the economical storage of enormous data and content, a different file system known as IPFS can be used. IPFS stands for Inter Planetary File System, a distributed, decentralized system and a platform to store data and files with high integrity and resiliency. It synthesizes thriving concepts from previous peer-to-peer networks, as well as DHTs, BitTorrent, Git, and SFS. The contribution of IPFS is connecting evidenced techniques into one significant system than the sum of its components. IPFS presents a brand new



Authorized licensed use limited to: Univ of Calif Santa Barbara. Downloaded on May 17,2021 at 18:05:47 UTC from IEEE Xplore. Restrictions apply.

platform for writing and deploying large scale applications. It is a new system for distributing and versioning massive data which can be used to spread computation and storage across multiple nodes in the network.

Fundamentally, IPFS is a peer-to-peer, open-sourced globally distributed classification system that can be used for storing and sharing massive volumes of files with high throughput [11]. Inter Planetary File System or IPFS is a protocol and a network which is designed to form content-addressable, peer-to-peer hypermedia storing and sharing system in a distributed network. Since IPFS is peer-to-peer, no nodes are privileged. IPFS nodes store IPFS objects in native storage. Nodes connect and transfer these objects, which represent files and data structures. IPFS uses the hash of the content to identify and share it among the nodes. It integrates alternative technologies like GIT for version management and MerkelDAG data structure for storing data. IPFS integrates each of the advanced Merkle-DAG structure with the data-addressability of P2P file-sharing systems. The content is distributed over a peer-to-peer network IPFS seeks to connect all computing devices with each other directly. IPFS can even evolve the web itself. In some ways, IPFS is comparable to the World Wide Web (www); however, IPFS can be seen as one BitTorrent swarm, exchanging objects inside one git repository. In alternative words, IPFS provides a high-throughput, content-addressed block storage model, with content-addressed hyperlinks. IPFS combines a distributed hash table with incentivized block exchange and a self-certifying namespace. IPFS has no single point of failure, and hence data transit can not be tampered by anyone. Distributed Content Delivery saves bandwidth and prevents DDoS attacks that Hypertext Transfer Protocol (HTTP) struggles with. The filesystem can be accessed in many kinds of ways like FUSE or ”File system in Userspace” over HTTP. A local file will be added to the IPFS file system, making it available to the globe. Files are known by their hashes; thus, it’s cache-friendly. Any user who downloads the file additionally serves the data to the other users of the network.

The IPFS Protocol is split into a stack of sub-protocols which are assigned totally different functions such as identities managers, network managers, content-based routing, exchange protocols (BitSwap), a content-addressed data structure (Merkle DAG) [12], version control (Git), naming service, etc. IPFS is primarily aimed at replacing HTTPS, but it can become a universal file transfer protocol [13]. IPFS can become compatible with gaming networks when combined with Quick UDP Internet Connections (QUIC) network protocol at the transport layer.

### *E. NFT - Non Fungible Tokens*

Fungible is anything that is transferable. Crypto-currencies such as Ether are fungible tokens as every single token is identical in use and value. On the contrary, a Non-Fungible Token has a unique value and identity. Every NFT is non-dividable and non-mergeable [14]. The first token standard adapted by the Ethereum Blockchain is ERC-20 (Ethereum Request for Comment) which supports Fungible Tokens only [15]. ERC-20 token standard promoted many ICOs which are based on Ethereum Blockchain. ERCs are predefined rules developed using Smart Contracts for implementing token measures in Ethereum Blockchain. ERC-20 can define by providing the contract address and availability of tokens. Lack of support for Non Fungible Tokens is one of the significant drawbacks of ERC-20, which led to another token standard ERC-721.

```
contract ERC721 {
    function name ( ) constant returns (string name);
    function symbol ( ) constant returns (string symbol);
    function toeaISuppIy() constant returns (uint256 toeaISuppIy);
    function balanceof(address _owner) constant returns (uint balance);
    function ownerOE (uint256 tokenId) constant returns (address owner);
    function approve (address _ to, uint256 _tokenId);
    function takeOwnership(unit256 _tokenId);
    function transfer (address _Eo, uint256 tokenId);
    function tokenOEOwnerByIndex( address _owner, uint256 index) constant returns (uint tokenId);
    function tokenMeeadata(uint256 _tokenId) constant returns (string infourl);
    event Transfer(address indexed _from, address indexed _ to, uint256 _tokenId);
    event Approval(address indexed _owner, address indexed _ approved, uint256 _tokenId);
}
```

Fig. 3. ERC-20 - Soildity Functions

ERC-721 token standard supports Non Fungible Tokens. It makes these tokens have a unique value and identity [16]. Tokens are attached to digital objects using metadata to help off-chain rendering or storage. Limitations of ERC-721 include lack of support for multiple tokens in a single, smart contract. Games have various types of unique assets and to support numerous assets, several quick contacts have to be implemented, which increases the gas fee and latency.

### *F. ERC 1155*

ERC-1155 is the new final token standard on the Ethereum Blockchain [17]. It is a universal standard as it supports the features of ERC-20 (Fungible) and ERC-721 (Non-Fungible). It enables game objects to possess a real-world value as it can help multiple tokens in a single contract. Trades and minting of tokens are more comfortable with the introduction of ERC-1155. ERC 1155 also supports converting existing tokens and minting new tokens out of them. Though the focus of this paper is limited to gaming, ERC-1155 has applications in several areas such as documentation and artwork.

```
contract ERC1155 is IERC1155, ERC165, CommonConstants
{
    function safeTransferFrom(address _from, address _to, uint256 _id, uint256 _value, bytes calldata _data);
    function safeBatchTransferFrom(address _from, address _to, uint256[] calldata _ids, uint256[] calldata _values, bytes calldata _data);
    function balanceOf(address _owner, uint256 _id);
    function balanceOfBatch(address[] calldata _owners, uint256[] calldata _ids);
    function setApprovalForAll(address _operator, bool _approved);
    function isApprovedForAll(address _owner, address _operator);
    function _doSTACheck(address _operator, address _from, address _to, uint256 _id, uint256 _value, bytes memory _data);
    function _doSBACheck(address _operator, address _from, address _to, uint256[] memory _ids, uint256[] memory _values, bytes memory _data);
}
```

Fig. 4. ERC-1155 - Soildity Functions

## III. Proposed Architecture

The proposed architecture enables decentralized computation and token infrastructure for gaming networks. It integrated IPFS and Ethereum blockchain to distribute the computation requirements in the game networks without any centralized agency. Independent nodes can provide computational activities such as rendering by utilizing their local machines in exchange for rewards which are supported



Authorized licensed use limited to: Univ of Calif Santa Barbara. Downloaded on May 17,2021 at 18:05:47 UTC from IEEE Xplore. Restrictions apply.

by Ethereum. This incentivizes more nodes to participate in the network. The decentralized token infrastructure integrates the ERC 1155 protocol to generate unique game objects which have a real-world value. These objects can be traded for other objects or currency and can also be minted into new objects. The proposed infrastructure is divided into following four elements:

*1) Proxy Computation:* Decentralized computing is an essential aspect of the proposed architecture. Computing is distributed among several nodes in the network instead of relying on a centralized game server. This completely eliminates the need for central gatekeepers in the gaming networks. Proxies are any nodes willing to participate in the network by utilizing their computational resources. In return for the computational work performed, proxies receive rewards that are based on Ethereum. This incentivizes proxies to participate in the network. In the case of multiplayer games, the computational activities include real-time transmission of data about the game state to the player nodes. In the case of single-player games, proxies act as distributed rendering nodes by effectively replacing the slow and inefficient cloud gaming servers.

*2) Data Transfer (IPFS + QUIC):* The consensus is an important aspect when it comes to game networks. All the players should have a consensus regarding the state of the game. The proposed architecture utilizes IPFS for transferring data related to states of the nodes in the network. IPFS eliminates the need for any centralized game server as the nodes can directly communicate with each other. The data related to the state of the nodes is exchanged using the Bitswap protocol. IPFS supports a wide range of networking protocols such as aa QUIC and UDP, which are being actively used in the existing gaming networks.

The steps involved in the computation are:

1) Game Networks have a pool of players producing data related to their states. In a multiplayer game, these states have to be synced with each other off-chain, and in a single-player, game rendering has to be performed off-chain. Hence gamers directly send the data related to their states onto a pool of proxies through IPFS.

2) Proxies receive the data related to the states and communicate with other proxies in the network to sync all the players' states. In the case of single-player games, proxies receive the data related to the states and render the game content on their machines.

3) Proxies transfer the synced and rendered data back to the players using the IPFS.

4) Players can also perform in-game communication among themselves as they are directly connected.

5) All the transactions between the nodes are recorded on the Ethereum blockchain after being validated by game smart-contracts.

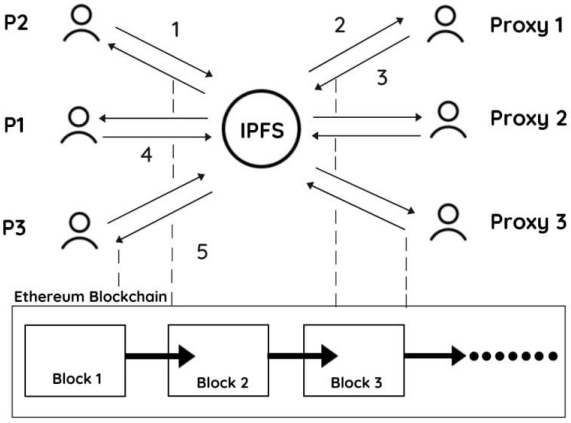


Fig. 5. Architecture - Computation

*3) Decentralized NFT Management:* The proposed architecture utilizes ERC-1155 extensively as the universal token standard. Each and every token is based on ERC-1155 as this will give consistency across all players and games.

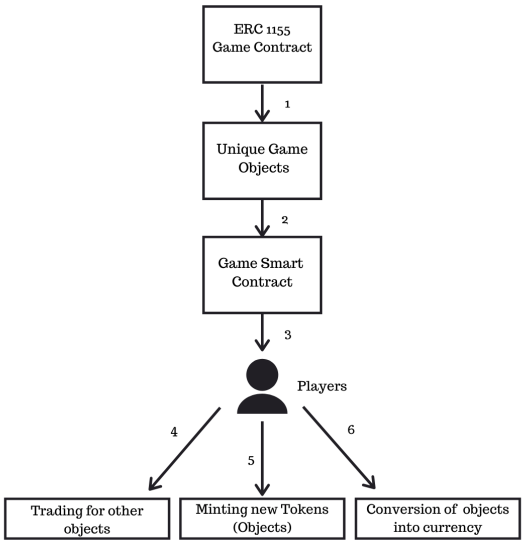


Fig. 6. Architecture - NFT Distribution

These steps involved in the NFT distribution are:

1) The creators of the game will frame the ERC-1155 contract concerning the game and attach them to the game objects present in the game using metadata.

2) The objects are rendered in the game based on the rules set in the game smart-contract.

3) Players receive objects in the game, and objects are added to their inventory, based on the Ethereum wallet.



Authorized licensed use limited to: Univ of Calif Santa Barbara. Downloaded on May 17,2021 at 18:05:47 UTC from IEEE Xplore. Restrictions apply.

4) Tokens received by the player can be traded with other players for other objects by exchanging them as ERC-1155 supports NFT transfers. The game object attached to the NFT is also transferred as the metadata is attached.

5) Players can mint the NFT by removing the metadata attached to the token and attach metadata related to another game object to it.

6) Players can also convert their NFTs to cryptocurrency directly without depending on any third party market places.

*4) Fraud Detection:* Gaming networks are prone to attacks and cheats every time. In order to protect these networks from attacks, a fail-proof monitoring mechanism can be deployed. In the proposed architecture, the nodes communicate using the IPFS protocol integrated with Ethereum Blockchain. Every network transaction performed by the player is recorded over the blockchain, as it maintains an immutable distributed ledger among all of its participants. Anomalies and attacks can be detected easily, as they are validated and seen by the entire network. In this way, fraudulent nodes can be eliminated from the network, which enables fair play. A private blockchain network can be used in order to reduce the latency in the network.

## IV. Experimental Result

This section focuses on the evaluation of the proposed decentralized gaming architecture and its efficiency. Relying on the distributed proxy computation architecture reduced the latency and improved reliability in the gaming network. The below metric shows a comparison between Game Server and Proxy Computation.

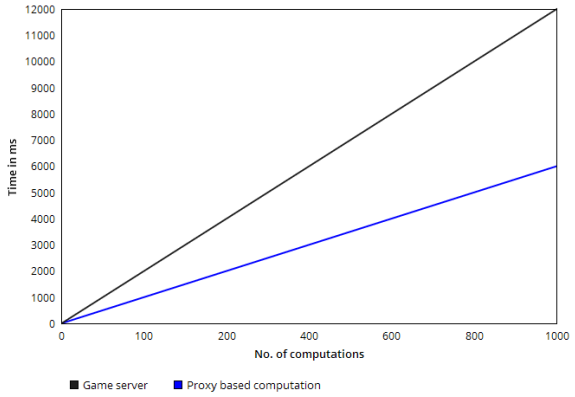


Fig. 7. Game Server vs Proxy Computation

## V. Conclusion

The proposed architecture utilizes the decentralized network of Etherereum Blockchain which ensures the availability of nodes. In a centralized scenario, the availability of the game depends on the availability of the server. Whereas in a decentralized scenario there are independent proxy nodes that are available throughout. Although for it to be scalable the network should be large to an extent where there are enough populated proxy nodes.

The vision of this paper is to propose a fully decentralized gaming infrastructure. This paper also discussed the difficulties with the current centralized gaming networks and proposed a protocol for its complete decentralization. As the proposed protocol is a proof of concept, it is advised not to use it for production. Reliability on Ethereum 1.0 can slow down the network as it has a consensus mechanism based on proof of work. But this can be improved by integrating other proof of stake based blockchains. The smart contract programming language solidity is a fully functional programming language which can cause security issues with the NFTs, usage of other languages such as Vyper is preferred.

## References

[1] K. Anderton, "The business of video games: Market share for gaming platforms in 2019 [infographic]," Jun 2019. [Online]. Available: https://www.forbes.com/sites/kevinanderton/2019/06/26/the-business-of-video-games-market-share-for-gaming-platforms-in-2019-infographic/6f39edfe7b25

[2] C. Gough, "Number of gamers worldwide 2021," Aug 2019. [Online]. Available: https://www.statista.com/statistics/748044/number-video-gamers-world/

[3] C. D. Merwin, "esports joins the big leagues," 2018. [Online]. Available: https://www.goldmansachs.com/insights/pages/infographics/e-sports/

[4] B. Ward, Y. Khmelevsky, G. Hains, R. Bartlett, A. Needham, and T. Sutherland, "Gaming network delays investigation and collection of very large-scale data sets," *2017 Annual IEEE International Systems Conference (SysCon)*, 2017.

[5] M. Claypool and D. Finkel, "The effects of latency on player performance in cloud-based games," *2014 13th Annual Workshop on Network and Systems Support for Games*, 2014.

[6] B. Mariano and S. G. M. Koo, "Is cloud gaming the future of the gaming industry?" *2015 Seventh International Conference on Ubiquitous and Future Networks*, 2015.

[7] D. Uribe, "Privacy laws, non-fungible tokens, and genomics," *The Journal of The British Blockchain Association*, vol. 3, no. 2, p. 1–10, 2020.

[8] V. Buterin, "Ethereum: A next-generation smart contract and decentralized application platform," Dec 2014.

[9] Nick, "Smart contracts: Building blocks for digital markets," *Organization of Phonetic Sciences, Amsterdam*, 1996. [Online]. Available: https://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/CDROM-/Literature/LOTwinterschool2006/szabo.best.vwh.net/-smartcontracts2.html

[10] R. Norvill, B. B. F. Pontiveros, R. State, and A. Cullen, "Visual emulation for ethereums virtual machine," *NOMS 2018 - 2018 IEEE/IFIP Network Operations and Management Symposium*, 2018.

[11] J. Benet, "Ipfs - content addressed, versioned, p2p file system," Jul 2014. [Online]. Available: https://ipfs.io/ipfs/QmR7GSQM93Cx5eAg6a6yRzNde1FQv7uL6X1o4k-7zrJa3LX/ipfs.draft3.pdf

[12] A. Auvolat and F. Taiani, "Merkle search trees: Efficient state-based crdts in open networks," *2019 38th Symposium on Reliable Distributed Systems (SRDS)*, 2019.

[13] Kingma, F. H., Abbeel, Pieter, and Jonathan, "Bit-swap: Recursive bits-back coding for lossless compression with hierarchical latent variables," Oct 2019. [Online]. Available: https://arxiv.org/abs/1905.06845

[14] S. Chevet, "Blockchain technology and non-fungible tokens: Reshaping value chains in creative industries," *SSRN Electronic Journal*, 2018.

[15] V. Buterin and F. Vogelsteller, "Eip 20: Erc-20 token standard," Nov 2015. [Online]. Available: https://eips.ethereum.org/EIPS/eip-20

[16] W. Entriken and D. Shirley, "Eip 721: Erc-721 non-fungible token standard," Jan 2018. [Online]. Available: https://eips.ethereum.org/EIPS/eip-721

[17] W. Radomski, A. Cooke, and P. Castongua, "Eip 1155: Erc-1155 multi token standard," Jun 2018. [Online]. Available: https://eips.ethereum.org/EIPS/eip-1155



Authorized licensed use limited to: Univ of Calif Santa Barbara. Downloaded on May 17,2021 at 18:05:47 UTC from IEEE Xplore. Restrictions apply.