



HAMMER CHAIN WHITEPAPER

VER. 0.2
2021.09

TABLE OF CONTENTS

Chapter 1. Overview

Chapter 2. HAMMER Fundamentals

1. What is HAMMER?00
 - 1.1 What makes HAMMER valuable?
 - 1.2 What is HAMMER Blockchain?
 - 1.3 How does HAMMER work?
 2. Smart Contract
 - 2.1 What is a Smart Contract?
 - 2.2 Key features
 3. HAMMER Mining
 - 3.1 Mining Process & Mining Pool
 - 3.2 Proof of Work(PoW)
 - 3.3 HMC Nodes
 4. HAMMER Gas
 5. HAMMER Coin
 - 5.1 HMC-20
 - 5.2 HMC-721
-

Chapter 3. HAMMER Ecosystem

1. Ecosystem Components
 - 1.1 Cryptocurrency Exchange - MetaEarthEX
 - 1.2 Trading Bot(App) – Alephrobot
 - 1.3 Wallet - HAMMER Mining
 - 1.4 SNS - MEMETOK
 - 1.5 DApp Development
 - 1.6 Minting NFT
 2. HAMMER Reward system
 3. Roadmap
-

Chapter 4. Legal Disclaimer

CHAPTER 1

Overview

HAMMER is a blockchain project launching and operating a network, in other words, a Mainnet. Transaction processing between the cryptocurrency exchanges and personal wallets is an independent wallet that forms an ecosystem and cryptocurrency wallets.

Also, HAMMER is a decentralized computing platform that holds its Mainnet protocol. HAMMER's main idea is to allow developers to make codes that are processed and performing through decentralized networks.

HAMMER used a language for extension prepared with Turing-completeness such as Solidity to allow developers to create arbitrary rules, transaction format, and state-transition functions. This also allowed them integrating into different sectors by typing Smart Contract for extending to various business areas using blockchain technology.

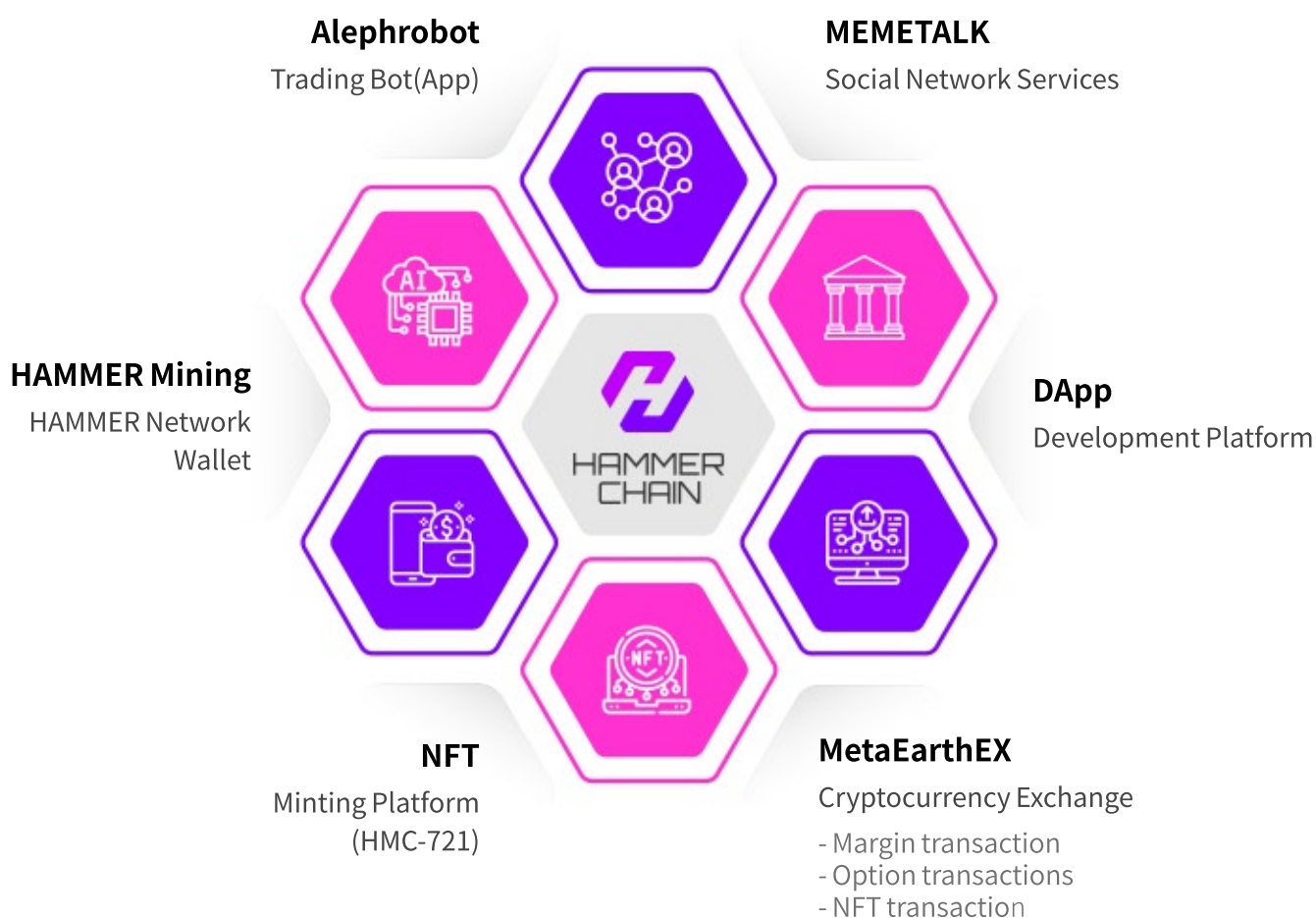


Figure01. HAMMER Project Overview

HAMMER is a decentralized application establishing platform. HAMMER's main characteristic is allowing developers to set specific parameters related to tokens as rules such as issued quantity, issuing methods, division possibility, and replacement possibility are set on the Smart Contract.

The technical specification of the HAMMER mainnet token is HMC-20 and the NFT issuance standard is HMC-721. The tokens and NFT issued by HAMMER blockchain consider HMC-20 standard and HMC-721 respectively. HMC-721 is an extended version of one of the most used NFT standards called ERC-721 and is exchangeable with the EVM (Ethereum Virtual Machine).

Self-issued HAMMER Coin (HMC) by HAMMER is a fuel that maintains the HAMMER platform ecosystem and a basic payment method for all transactions in the platform. The gas fee that occurred on the HAMMER blockchain also must be paid with HMC.

By repeating the process of finding a hash less than the desired values, HMC is created through a Proof of Work (PoW) method of mining, the consensus algorithm that proves it participated in the specific work. Later on, HAMMER will plan to change into the Proof of Stake (PoS) method, a consensus algorithm that gives decision-making authority in proportion to the cryptocurrency holding share ratio.

HAMMER ecosystem is composed of a global cryptocurrency exchange known as MetaEarthEX, a trading bot Alephrobot, the HAMMER network wallet service HAMMER Mining and public social network service (SNS) MEMETOK and is currently on service in WEB 2.0 environment. Works are in the process to add services of decentralized application (DApp) development and NFT (Non-Fungible Token) minting in time for launching the HAMMER mainnet.

CHAPTER 2

HAMMER Fundamentals

1. What is HAMMER?

HAMMER is a decentralized computing platform that holds its Mainnet protocol. The main idea of HAMMER allows developers to create and process the code through a decentralized network, not on the centralized server.

HAMMER is a decentralized autonomous blockchain network comprised of Developers and Miners. The developers create the code and implement technical changes to the network. The miners help generate the new coins, confirm transactions and contribute to the governance of the blockchain.

And HAMMER is a platform that hosts smart contracts. Smart contracts are pieces of code that execute automatically when a specific set of conditions is met. They utilize HAMMER’s global computing network. HAMMER also provides a decentralized digital currency called HAMMER Coin(HMC) to facilitate value exchange within the network or cryptocurrency exchanges.

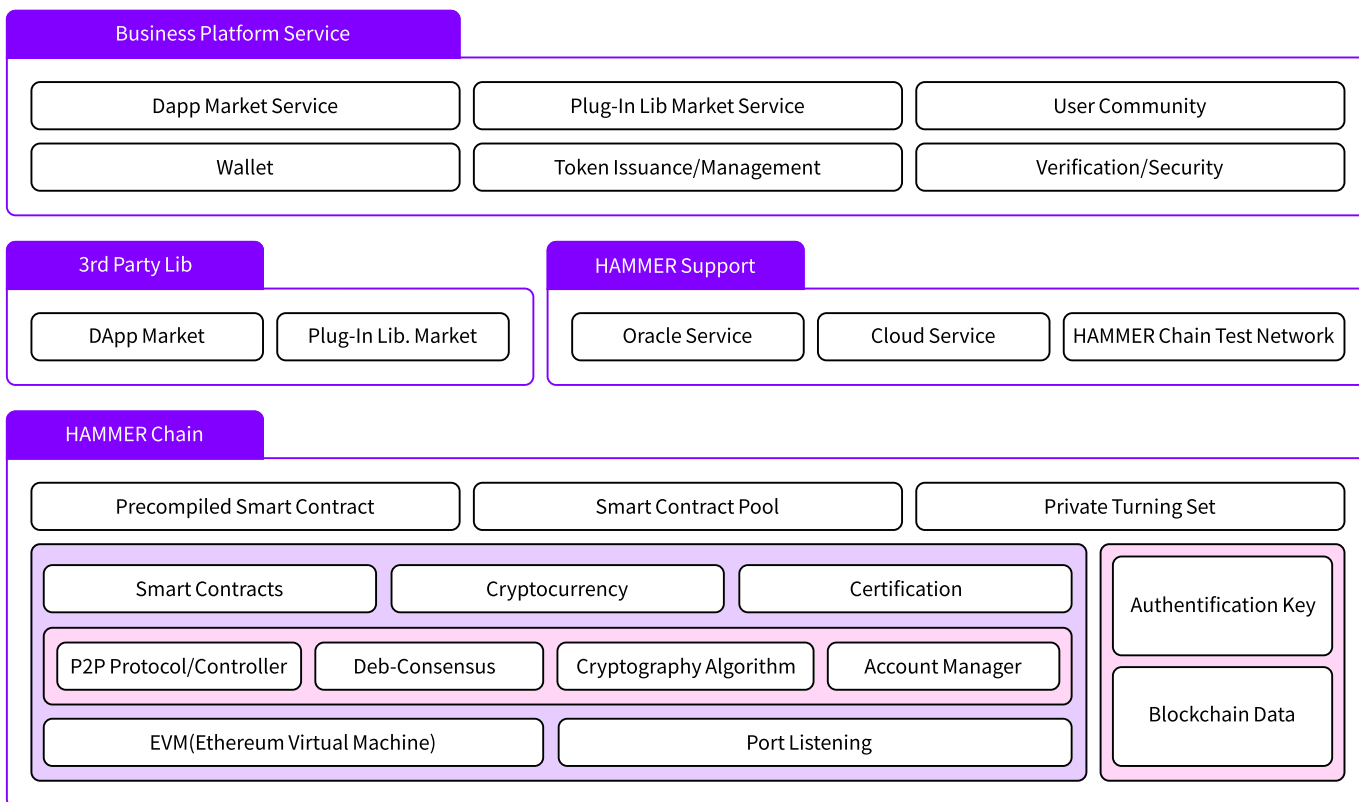


Figure02. HAMMER Blockchain Structure

1.1 What makes HAMMER valuable?

We touched on the idea that HAMMER can run code across a distributed system. As such, programs can’t be tampered with by external parties. They’re added to HAMMER’s database (i.e., the blockchain), and can be programmed so that the code can’t be edited. In addition, the database is visible to everyone, so users can audit code before interacting with it. What this means is that anyone, anywhere, can launch applications that can’t be taken offline. More interestingly, because its native unit – HMC – stores value, these applications can

set conditions on how value is transferred. We call the programs that makeup applications smart contracts. In most cases, they can be set to operate without human intervention.

1.2 What is HAMMER Blockchain?

The blockchain lies at the heart of HAMMER – it’s the database that holds the information used by the protocol. It helps to think of HAMMER’s blockchain as a book that you keep adding pages to. Each page is called a block, and it’s filled with information about transactions.

When we want to add a new page, we need to include a special value at the top of the page. This value should allow anyone to see that the new page was added after the previous page, and not just inserted into the book randomly. For this process, a process called hashing is used. Hashing takes a piece of data (everything on our page) and returns a unique identifier (or hash).

The odds of two pieces of data giving us the same hash are astronomically low. You can easily calculate a hash, but you can’t reverse the hash to get the information used to create it. We’ll get into why this is important for mining in a later chapter.

1.3 How does HAMMER work?

The smart contracts that run on HAMMER are triggered by transactions. When a user sends a transaction to a contract, every node on the network runs the contract’s code and records the output. It does this by using the Ethereum Virtual Machine (EVM), which converts the smart contracts into instructions the computer can read.

To update the state, a special mechanism called mining is used (for now). Mining is done with a Proof of Work algorithm, much like Bitcoin’s. We’ll get into more depth on this shortly.

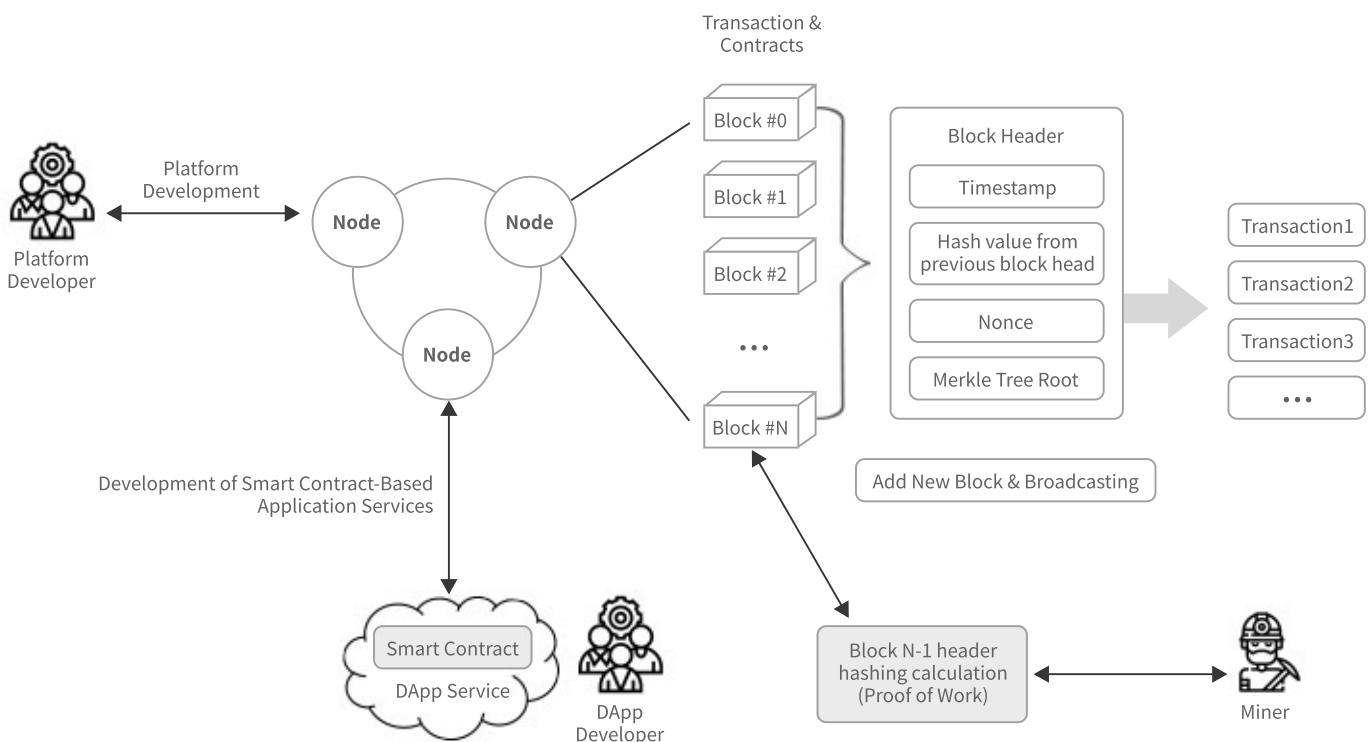


Figure03. HAMMER Blockchain Work Process

2. Smart Contract

A smart contract is a type of code, but it executes the contract by itself in a predetermined condition and it can be seen as a contract in the sense that it forces the contract between the parties. Developers in HAMMER determine the code to allow EVM to read and send the contract to the specific address to implement the contract details.

2.1 What is a Smart Contract?

In the world of cryptocurrencies, we may define a smart contract as an application or program that runs on a blockchain. Typically, they work as a digital agreement that is enforced by a specific set of rules. These rules are predefined by computer code, which is replicated and executed by all network nodes.

Through a smart contract, you can make a contract containing the promising contract details of both parties without forming a trust relationship or not need to know. The smart contract would then be implementing a specific work when it fulfills specific conditions or would not implement the contract when it does not meet the condition.

On the HAMMER network, smart contracts are responsible for executing and managing the blockchain operations that take place when users (addresses) interact with each other. Thus, smart contracts are controlled by computer code, and EOAs are controlled by users. Any address that is not a smart contract is called an externally owned account (EOA).

HAMMER smart contracts are made of a contract code and two public keys. The first public key is the one provided by the creator of the contract. The other key represents the contract itself, acting as a digital identifier that is unique to each smart contract. The deployment of any smart contract is made through a blockchain transaction, and it can only be activated when called by an EOA (or by other smart contracts). However, the first trigger is always caused by an EOA (user).

2.2 Key features

A HAMMER smart contract presents the following characteristics:

- Distributed

Smart contracts are replicated and distributed in all nodes of the HAMMER network. This is one of the major differences from other solutions that are based on centralized servers.

- Deterministic

Smart contracts only perform the actions they were designed to, given the requirements are met. Also, the outcome will always be the same, no matter who executes them.

- Autonomous

Smart contracts can automate all sorts of tasks, working like a self-executing program. In most cases, though, if a smart contract isn't triggered, it will stay "dormant" and won't perform any action.

- Immutable

Smart contracts can't be changed after being deployed. They can only be "deleted" if a particular function was previously implemented.

- Customizable

Before deployment, smart contracts can be coded in many different ways. So, they can be used to create many types of decentralized applications (DApps). This is related to the fact that HAMMER is a Turing complete blockchain.

※ Turing Completeness of blockchain

Turing-Completeness of blockchain can be obtained through Turing Complete Language. Turing Complete Language is based on infinite storage space creating a Turing Machine, a machine that can solve all problems in the world and a language that implements algorithms that can be entered into Turing Machine. HAMMER provides a programming language called 'Solidity' and introduced the concept of 'Gas' to solve the Turing Incomplete of the script language. It then secured Turing Complete of blockchain by making it happen on a system basis that charges gas for each computer code operation. Due to this Turing Completeness, various decentralized applications (DApps) can be created based on the HAMMER platform.

- Trustless

Two or more parties can interact via smart contracts without knowing or trusting each other. In addition, blockchain technology ensures that data is accurate.

- Transparent

Since smart contracts are based on a public blockchain, their source code is not only immutable but also visible to anyone.

HAMMER smart contract does not allow add new functions after the establishment. However, when the crater adds a SELFDESTRUCT function on the code, you can delete the smart contract and exchange it for a new contract. In contrast, you cannot delete the functions that were not included in the code in advance.

Especially, for the upgradeable smart contract, developers allow flexibility for the invariability of the contract. In the case of smart contracts divided into several smaller contracts, if some parts were designed as unchangeable and the other parts are activated with the delete function, then the latter ones can be alternated or deleted however, other functions stay as it is.

Programmable smart contracts have strengths that it provides different types of service and solution as users can design it in various ways. As a decentralized and self-adapting program, smart contract reduces the operational cost and increase transparency. It is usable in the condition of transferring or exchanging the funds between both parties and is designable according to the cases.

Since smart contracts are made with human-made computer code, some vulnerable points or risks of occurring bugs follow. Importantly for the case of accompanying important information or a huge amount of fees, it is ideal for skilled programmers to write or batch.

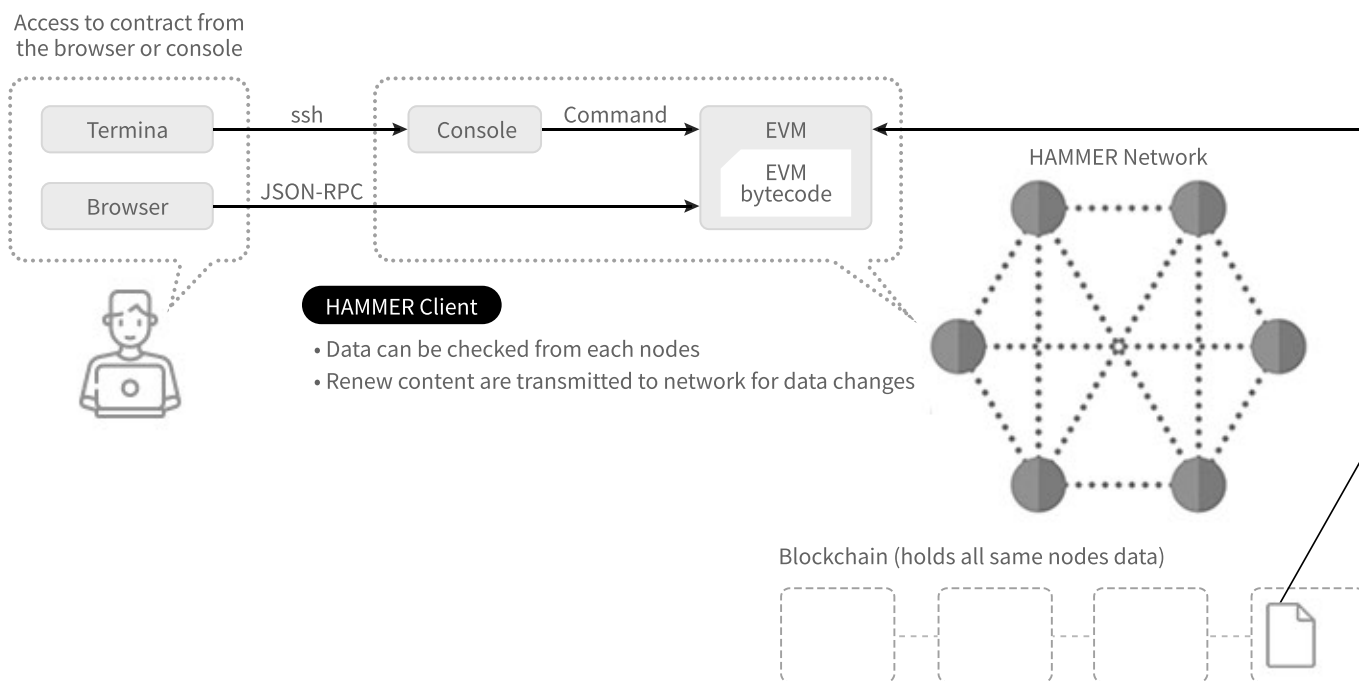


Figure04. How Smart Contract Works

3. HAMMER Mining

The mining process is integral to securing and updating the blockchain. In HAMMER, the same principle holds: to reward the users that mine (which is costly), the protocol rewards them with HMC. Mining is critical to the security of the network. It ensures that the blockchain can be updated fairly and allows the network to function without a single decision-maker.

What they're doing is hashing a set of pending transactions alongside some other data. For the block to be considered valid, the hash needs to fall below a value that's set out by the protocol. If they're unsuccessful, they can modify some of the data and try again. To compete with others, miners, therefore, need to be able to hash as fast as possible – we measure their power in hash rate.

The more hash rate there is on the network, the harder the puzzle becomes to solve. As you can imagine, continuously hashing at high speeds is expensive. To incentivize miners to secure the network, they earn a reward receive HMC.

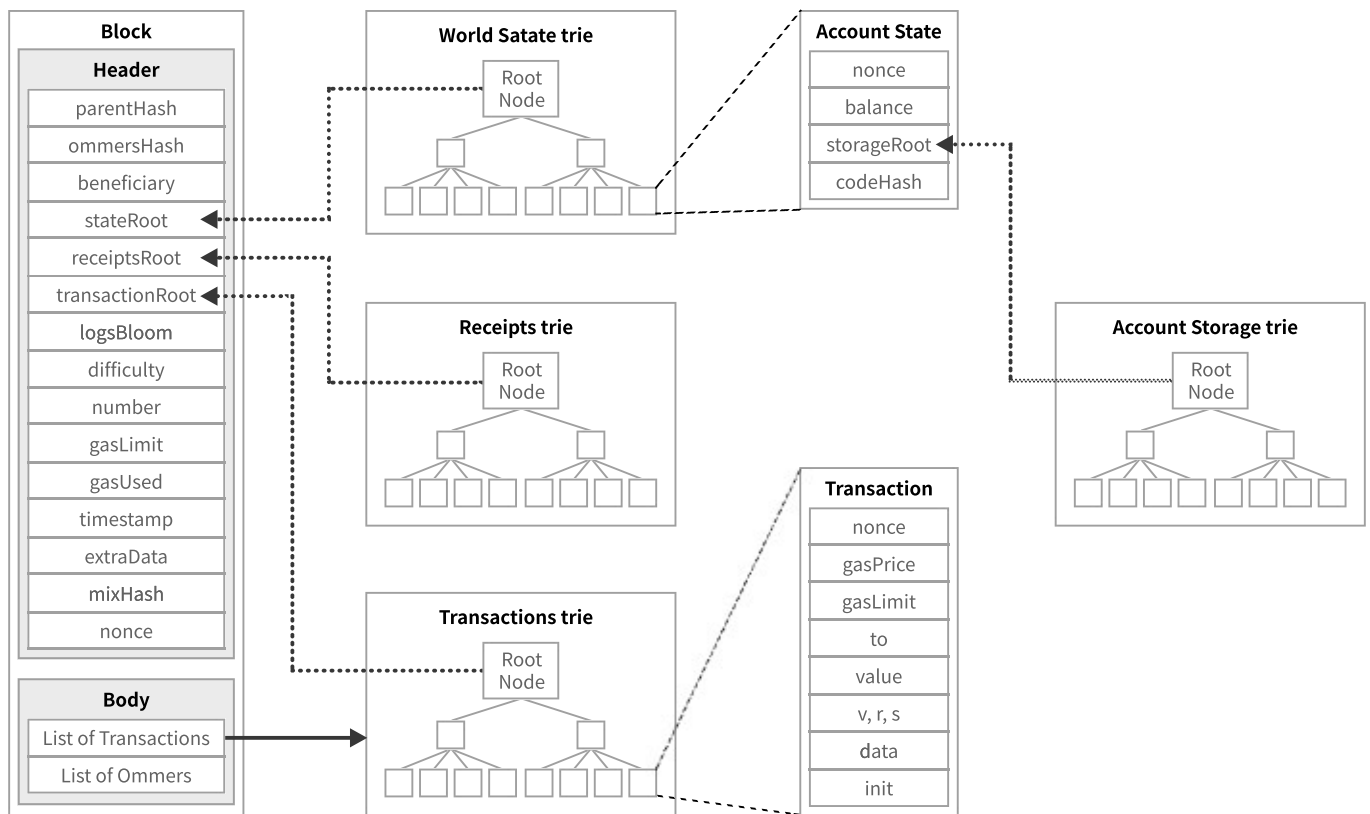


Figure05. Block Header Structure

3.1 Mining Process & Mining Pool

Cryptocurrency mining is the process in which transactions between users are verified and added to the blockchain public ledger. The process of mining is also responsible for introducing new coins into the existing circulating supply and is one of the key elements that allow cryptocurrencies to work as a peer-to-peer decentralized network, without the need for a third-party central authority.

A miner is a node in the network that collects transactions and organizes them into blocks. Whenever transactions are made, all network nodes receive them and verify their validity. Then, miner nodes gather these transactions from the memory pool and begin assembling them into a block (candidate block).

The first step of mining a block is to individually hash each transaction taken from the memory pool, but before starting the process, the miner node adds a transaction where they send themselves the mining reward (block reward). This transaction is referred to as the coin base transaction, which is a transaction where coins get created 'out of thin air' and, in most cases, is the first transaction to be recorded in a new block.

After every transaction is hashed, the hashes are then organized into something called a Merkle Tree (or a hash tree) - which is formed by organizing the various transaction hashes into pairs and then hashing them. The outputs are then organized into pairs and hashed once again, and the process is repeated until "the top of the tree" is reached. The top of the tree is also called a root hash (or Merkle root) and is a single hash that represents all the previous hashes that were used to generate it.

The root hash - along with the hash of the previous block and a random number called nonce - is then placed into the block's header. The block header is then hashed producing an output based on those elements (root hash, previous block's hash, and nonce) plus a few other parameters. The resulting output is the block hash and will serve as the identifier of the newly generated block (candidate block).

To be considered valid, the output (block hash) must be less than a certain target value that is determined by the protocol. In other words, the block hash must start with a certain number of zeros.

The target value - also known as the hashing difficulty - is regularly adjusted by the protocol, ensuring that the rate at which new blocks are created remains constant and proportional to the amount of hashing power devoted to the network.

Therefore, every time new miners join the network and competition increases, the hashing difficulty will raise, preventing the average block time from decreasing. In contrast, if miners decide to leave the network, the hashing difficulty will go down, keeping the block time constant even though there is less computational power dedicated to the network.

The process of mining requires miners to keep hashing the block header over and over again, by iterating through the nonce until one in the network miner eventually produces a valid block hash. When a valid hash is found, the founder node will broadcast the block to the network. All other nodes will check if the hash is valid and, if so, add the block into their copy of the blockchain and move on to mining the next block.

However, it sometimes happens that two miners broadcast a valid block at the same time and the network ends up with two competing blocks. Miners start to mine the next block based on the block they received first. The competition between these blocks will continue until the next block is mined based on either one of the competing blocks. The block that gets abandoned is called an orphan block or a stale block. The miners of this block will switch back to mining the chain of the winner block.

While the block reward is granted to the miner who discovers the valid hash first, the probability of finding the hash is equal to the portion of the total mining power on the network. Miners with a small percentage of the mining power stand a very small chance of discovering the next block on their own. Mining pools are created to solve this problem. It means pooling of resources by miners, who share their processing power over a network, to split the reward equally among everyone in the pool, according to the amount of work they contribute to the probability of finding a block.

3.2 Proof of Work (PoW)

Proof of Work (commonly abbreviated to PoW) is a mechanism for preventing double-spends. Most major cryptocurrencies use this as their consensus algorithm. That’s just what we call a method for securing the cryptocurrency’s ledger.

3.2.1 Why is Proof of Work necessary?

The blockchain is a big database that every user can see, so they can check if funds have been spent before. There’s another intricacy here – each time you make a transaction, you refer to the transaction where the funds came from. Now, we have a way to track the units. The group won’t allow the transaction to be added to the notepad.

This is where Proof of Work comes in. It ensures that users aren’t spending money that they don’t have the right to spend. By using a combination of game theory and cryptography, a PoW algorithm enables anyone to update the blockchain according to the rules of the system.

3.2.2 How does PoW work?

Our notepad above is the blockchain. But we don’t add transactions one by one – instead, we lump them into blocks. We announce the transactions to the network, then users creating a block will include them in a candidate block. The transactions will only be considered valid once their candidate block becomes a confirmed block, meaning that it has been added to the blockchain.

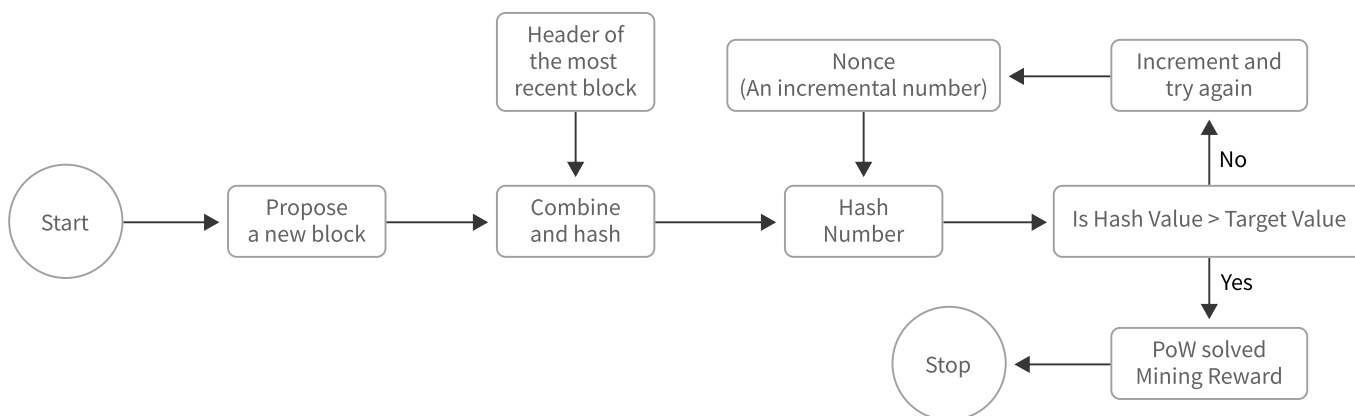


Figure06. Proof of Work Flowchart

Appending a block isn't cheap, however. Proof of Work requires that a miner (the user creating the block) uses up some of their resources for the privilege. That resource is computing power, which is used to hash the block's data until a solution to a puzzle is found.

Hashing the block's data means that you pass it through a hashing function to generate a block hash. The block hash works like a "fingerprint" – it's an identity for your input data and is unique to each block.

It's virtually impossible to reverse a block hash to get the input data. Knowing an input, however, it's trivial for you to confirm that the hash is correct. You just have to submit the input through the function and check if the output is the same.

In Proof of Work, you must provide data whose hash matches certain conditions. Your only option is to pass your data through a hash function and to check if it matches the conditions. If it doesn't, you'll have to change your data slightly to get a different hash. Changing even one character in your data will result in a different result, so there's no way of predicting what an output might be.

As a result, if you want to create a block, you're playing a guessing game. You typically take information on all of the transactions that you want to add and some other important data, then hash it all together. But since your dataset won't change, you need to add a piece of information that is variable. Otherwise, you would always get the same hash as output. This variable data is what we call a nonce. It's a number that you'll change with every attempt, so you're getting a different hash every time. And this is what we call mining.

Summing up, mining is the process of gathering blockchain data and hashing it along with a nonce until you find a particular hash. If you find a hash that satisfies the conditions set out by the protocol, you get the right to broadcast the new block to the network. At this point, the other participants of the network update their blockchains to include the new block.

The higher the hash rate on the network, the more difficult it is to find a valid hash. This is done to ensure that blocks aren't found too quickly.

As you can imagine, trying to guess massive amounts of hashes can be costly on your computer. You're wasting computational cycles and electricity. But the protocol will reward you with cryptocurrency if you find a valid hash.

What's to stop you from putting a bunch of fraudulent transactions into the block and producing a valid hash?

That's where public-key cryptography comes in. What is Public-Key Cryptography? We use some neat cryptographic tricks that allow any user to verify whether someone has a right to move the funds they're attempting to spend.

When you create a transaction, you sign it. Anyone on the network can compare your signature with your public key, and check whether they match. They'll also check if you can spend your funds and that the sum of your inputs is higher than the sum of your outputs (i.e., that you're not spending more than you have). Any block that includes an invalid transaction will be automatically rejected by the network.

Therein lies the beauty of Proof of Work: it makes it expensive to cheat, but profitable to act honestly. Any rational miner will be seeking ROI, so they can be expected to behave in a way that guarantees revenue.

Proof of Work was the original solution to the double-spend problem and has proven to be reliable and secure. With clever use of cryptography, hash functions, and game theory, participants in a decentralized environment can agree on the state of a financial database.

3.3 HAMMER Nodes

“HAMMER node” is a term that can be used to describe a program that interacts with the HAMMER network in some way. A HAMMER node can be anything from a simple mobile phone wallet application to a computer that stores an entire copy of the blockchain.

3.3.1 HAMMER Full Nodes

To interface with the HAMMER network in a way that allows you to validate blockchain data independently, you need to run a full node using software like the ones mentioned above.

The software will download blocks from other nodes and verify if the transactions included are correct. It will also run all of the smart contracts that have been called to ensure that you’re receiving the same information as other peers. If all is working as intended, we can expect every node to have an identical copy of the blockchain on their machines.

Full nodes are vital to the functioning of HAMMER. Without multiple nodes spread around the globe, the network would lose its censorship-resistant and decentralized properties.

3.3.2 HAMMER Light Nodes

Running a full node allows you to contribute directly to the health and security of the network. But a full node often requires a separate machine to operate as well as occasional maintenance. Light nodes might be a better option for the users that are unable to run a full node (or that simply prefer not to do it). As the name might suggest, light nodes are lightweight – they use fewer resources and take up minimal space.

Light nodes are not entirely self-sufficient. They don’t sync the blockchain in full and therefore require full nodes to feed them relevant information. Light nodes are popular amongst merchants, services, and users. They’re used extensively for making and receiving payments in scenarios where full nodes are deemed unnecessary and too costly to run.

3.3.3 HAMMER Mining Nodes

To mine HAMMER, users need additional hardware. A common practice involves the construction of a mining rig. With these, users connect multiple GPUs (graphics processing units) to hash data at high speeds.

Miners have two options: mining solo, or in a mining pool. Solo mining means that the miner works alone to create blocks. If they’re successful, they do not share their mining rewards with anyone. Alternatively, when joining a mining pool, they combine their hashing power with that of other users. This will make them more likely to find a block, but they’ll also need to share their rewards with pool members.

4. HAMMER Gas

When a contract is executed as a result of being triggered by a message or transaction, every instruction is executed on every node of the network. This has a cost: for every executed operation there is a specified cost, expressed in some gas units.

The average gas fee will increase when the network jams and a large number of users try transactions, and when the activity is not too much, the gas fee will be reduced. All tasks need a certain amount of gas fee. This implies complex contract consumes more than simple transactions.

The price of gas is decided by the miners, who can refuse to process a transaction with a lower gas price than their minimum limit. To get a gas user simply need to add HMC to the user's account.

The HAMMER protocol charges a fee per computational step that is executed in a contract or transaction to prevent deliberate attacks and abuse on the HAMMER network. Every transaction is required to include a gas limit and a fee that it is willing to pay per gas.

Miners have the choice of including the transaction and collecting the fee or not. If the total amount of gas used by the computational steps spawned by the transaction, including the original message and any sub-messages that may be triggered, is less than or equal to the gas limit, then the transaction is processed.

If the total gas exceeds the gas limit, then all changes are reverted, except that the transaction is still valid, and the fee can still be collected by the miner. Not used gas is reimbursed to the sender as HMC. You are only charged for the gas you consume.

Just as you cannot drive a car without fuel, the contract cannot be done without gas. Contracts set gas amounts that need to be paid by users for successful operation. The contract terminates when it is insufficient for gas.

5. HAMMER Coin

A large part of HAMMER's appeal is the ability for users to create their own assets on-chain, which can be stored and transferred like HMC. The rules governing them are set out in smart contracts, allowing developers to set specific parameters regarding their tokens. These can include how many to issue, how to issue them, whether they're divisible, whether each is fungible, and many others.

Coin name	Ticker	Standard	Mainnet
HAMMER Coin	HMC	HMC-20	HAMMER

NFT Minting Authority	Standard	Mainnet
HAMMER Foundation	HMC-721	HAMMER

5.1 HMC-20

HMC-20 is a technical standard used to issue and implement tokens on the HAMMER blockchain. HAMMER is a platform for building decentralized applications, and as a tradeable token, HMC is the fuel of this ecosystem.

The HAMMER blockchain-issued tokens follow the HMC-20 standard. This standard explains a core function of all HAMMER-based tokens. HMC-20 is a technical standard that issues and implements tokens on the HAMMER blockchain. This standard explains a common rule set where tokens should follow to operate correctly in the HAMMER ecosystem. Therefore, HMC-20 should not be regarded as codes or software, but it can be explained as a technical manual or specification.

If the HMC-20 standard is used, developers can expect the interaction between application programs and different tokens more accurately. Also, it defines how HMC-20 tokens are transferred within the HAMMER blockchain or how each supply or address balance is recorded coherently.

In other words, HMC-20 provides a list of rules for developers to follow, and this allows active operation in the bigger HAMMER platform. If many DApps (Distributed Applications) and services are supporting HMC-20 tokens, the community members and corporations may use or adapt it easily.

HMC-20 compatible tokens that are established in the HAMMER blockchain can uniquely materialize. Depending on the accessing method, it presents various types of digital assets or transaction available products such as coins, vouchers, gold certificates, and loyalty points. HMC-20 standard will reduce the required effort to issue the digital tokens.

HMC-20 standard is technically explaining the partial functions and maintaining functions of HAMMER-based digital tokens. These functions include the token address, symbol, names, and token smart contract as well as its related important data transmitting method.

5.1.1 How to buy HMC with a credit/debit card

MetaEearthEX allows you to seamlessly buy HMC in your browser. To do so:

- Go to the Buy and Sell Cryptocurrency portal.
- Select the cryptocurrency you want to buy (HMC), and the currency you wish to pay with.
- Log in to MetaEearthEX, or register if you don't have an account.
- Select your payment method.
- If prompted, enter your details and complete the identity verification process.

- Go to the Buy and Sell Cryptocurrency portal.
- Select the cryptocurrency you want to buy (HMC), and the currency you wish to pay with.
- Log in to MetaEearthEX, or register if you don't have an account.
- Select your payment method.
- If prompted, enter your details and complete the identity verification process.

5.1.2 How to buy HMC on peer-to-peer markets

You can also buy and sell HMC on peer-to-peer markets. This allows you to purchase coins from other users, directly from the MetaEearthEX mobile app.

- Launch the app and log in or register.
- Select One-click buy-sell, followed by the Buy tab in the top left corner of the interface.
- You'll be prompted with several different offers – tap Buy on the one you wish to go with.
- You can pay with other cryptocurrencies (the By Crypto tab) or fiat currency (the By Fiat tab).

5.1.3 How to deposit your HMC to MetaEearthEX

If you already have HMC and want to deposit it on MetaEearthEX, you can simply follow these quick steps:

- Log in to MetaEearthEX, or register if you don't already have an account.
- Go to your Spot Wallet and select Deposit.
- Select HMC from the coin list.
- Select the network and send your HMC to the corresponding address.

5.1.4 How to withdraw your HMC from MetaEearthEX

If you already have HMC and want to withdraw it from MetaEearthEX, you can simply follow these quick steps:

- Log in to MetaEearthEX.
- Go to your Spot Wallet and select Withdraw.
- Select HMC from the coin list.
- Select the network
- Insert the recipient's address and amount.
- Confirm the process via email.

5.1.5 What is HAMMER staking

In Proof of Work protocols, the security of the network is assured by miners. The miners won't cheat, as it would waste electricity and cause them to lose out on potential rewards. In Proof of Stake, there is no such game theory, and different cryptographic economy measures are in place to ensure network security. Validators must put forward a stake (meaning a token holding) to be eligible for validation. This is a set amount of HMC that's lost if the node attempts to cheat, or is slowly depleted if the node is unresponsive or offline. However, if the validator runs additional nodes, they stand to gain more rewards. The estimated minimum stake for HAMMER is 32 HMC per validator. This is set so high to render the cost of attempting a 51% attack extremely high.

5.2 HMC-721

HMC-721 is a token standard written for unique or irreplaceable tokens in the HAMMER blockchain. Most tokens can be replaced in the HAMMER blockchain, however, HMC-721 cannot be replaced.

HMC-721 is an extension of Ethereum's ERC-721, one of the most commonly used NFT standards, and is compatible with the Ethereum Virtual Machine (EVM). Each NFT is unique and is not interchanged with any other token. Therefore HMC-721 tokens allow you to tokenize the ownership of data and attach to it a unique identifier.

This aspect makes the token one of a kind and vastly different from HMC-20 tokens. With HMC-20, developers can create multiple identical tokens within one smart contract. However, with HMC-721, each token is assigned a different token ID.

Through these unique IDs, HMC-721 tokens can be used to represent collectible, non-fungible items. Users can trade and transfer these tokens based on their market value, which depends on the token's rarity or utility.

HMC-721 tokens can represent:

- Digital and Physical Art
- Collectibles
- In-game items
- Physical property and real-estate
- Lottery tickets

Like other tokens on HAMMER Mainnet, HMC-721 token transfers require HMC for gas fees. HMC-721 tokens can be created on HAMMER Mainnet. HMC-721 has a list of functions that govern how the token interacts with the Binance Smart Chain. Some of these are fairly standard and are also present in the HMC-20 standard:

- Name

Defines the HMC-721 token's name, which other contracts will identify it by

- Symbol

A shorter name for the token similar to a ticker symbol

- Balance Of

Shows the token balance of a specific address

- Total Supply

Defines the total number of tokens created.

- A set of basic ownership functions

A unique feature for NFTs is the metadata function

- Token Metadata

This function allows for the inclusion of metadata into a token used to link to an artwork file or another collectible aspect of the NFT. For example, each CryptoPunk has metadata contained within the NFT that points it to a particular punk in a grid of 10,000 punks. Although CryptoPunks don't use the

- Difference between HMC-20 token and HMC-721 token

The reason why HMC-20 tokens could become the tradeable token in exchanges is that it supports fungible cryptocurrencies. The meaning of Fungible would be easier if imagining the money.

For example, a \$1 note you have and another \$1 note that another person has the same value. The same applies to other people. Tokens issued with HMC-20 have a characteristic of 'fungible'. On the other hand, HMC-721 tokens have a 'non-fungible characteristic. In other words, HMC-721 issued tokens have a different value on each of them.

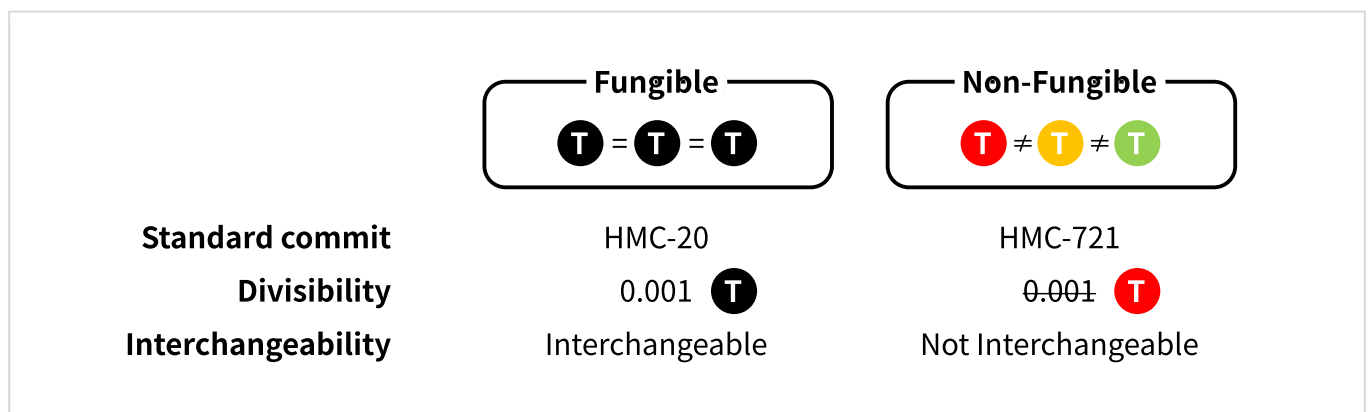


Figure07. Comparison of fungible and non-fungible tokens

- Applied technology

HMC-721 materializes a standard API of NFT in a smart contract and this standard provides basic functions that track and send the NFT. HMC-721 considered consignment about the third parties' middleman, wallet, and auctioneers as well as the NFT use cases. NFT is ownership of the digital and physical asset, it is identifiable and tracks each ownership.

- Physical property - houses, unique artwork
- Virtual collectables - unique pictures of kittens, collectable cards
- “Negative value” assets - loans, burdens and other responsibilities

HMC-721 standard interface allows wallet, broker, and auction applications to work with all NFT of HAMMER and provides a simple HMC-721 smart contract and NFT tracking contract.

You can investigate detailed information of NFT assets and the smart contract name through meta extension, and the enumerate extension shows a total list of NFT and allows searching in the contract.

- SOLIDITY ISSUE #3412

The above interfaces include explicit mutability guarantees for each function. Mutability guarantees are, in order weak to strong: payable, implicit non-payable, view, and pure. Your implementation MUST meet the mutability guarantee in this interface, and you MAY meet a stronger guarantee. For example, a payable function in this interface may be implemented as non-payable (no state mutability specified) in your contract.

- SOLIDITY ISSUE #3419

A contract that implements ERC721Metadata or ERC721Enumerable SHALL also implement ERC721. ERC-721 implements the requirements of interface ERC-165.

- SOLIDITY ISSUE #2330

If a function is shown in this specification as external, then a contract will be compliant if it uses public visibility.

NFT Identifiers

All NFTs are identified by a unique ID inside the smart contract. While some HMC-721 smart contracts may find it convenient to start with ID 0 and simply increment by one for each new NFT, callers shall not assume that ID numbers have any specific pattern to them, and MUST treat the ID as a “black box”. Also, note that an NFTs MAY become invalid (be destroyed)

Transfer Mechanism

HMC-721 standardizes a safe transfer function and unsafe function.

Transfer can be started with the following :

- The owner of an NFT
- The approved address of an NFT
- An authorized operator of the current owner of an NFT

Authorized operators can set authorized addresses for NFT. It provides powerful tools for wallet, broker, and auction applications, allowing you to use many NFTs quickly. The transfer and approval function document specify conditions only when a transaction must occur, except for a failed transaction. ERC-721 has only one unique NFT, so there is no tolerance.

Therefore, it has the advantages of the original design of ERC-20. When an operator sends a token, it becomes an operator that moves itself, not an operator that replaces the owner of the token. Therefore, the operator and the previous token owner are important to the token recipient.

Gas and Complexity

If your application can grow, then avoid using for/while loops in your code. These indicate your contract may be unable to scale and gas costs will rise over time without bound. Remove the asset enumeration function if it requires a for-loop, return a Solidity array type from enumeration functions.

Privacy

Wallets/brokers/auctioneers identified in the motivation section have a strong need to identify which NFTs an owner owns. It may be interesting to consider a use case where NFTs are not enumerable, such as a private registry of property ownership, or a partially private registry. However, privacy cannot be attained because an attacker can simply call the owner for every possible tokenId.

CHAPTER 3

HAMMER Ecosystem

1. Ecosystem Components

HAMMER ecosystem is composed with a global cryptocurrency exchange known as MetaEarthEX, a trading bot Alephrobot, the HAMMER network wallet service HAMMER Mining and public social network service (SNS) MEMETOK and is in a process to add services of decentralized application (DApp) development and NFT (Non-Fungible Token) minting in time for launching the HAMMER mainnet.

The technical specification of the HAMMER mainnet token is HMC-20. HAMMER is a decentralized application establishment platform, and HAMMER COIN (HMC) is fuel for maintaining the ecosystem and a basic payment method for all transactions in the platform.

Self-issued HAMMER Coin (HMC) by HAMMER is a fuel that maintains the HAMMER platform ecosystem and a basic payment method for all transactions in the platform. The gas fee that occurred on the HAMMER blockchain also must be paid with HMC.

HMC is created through a Proof of Work (PoW) method of mining, the consensus algorithm that proves it participated in the specific work. Later on, HAMMER will plan to change into the Proof of Stake (PoS) method, a consensus algorithm that gives decision-making authority in proportion to the cryptocurrency holding share ratio.

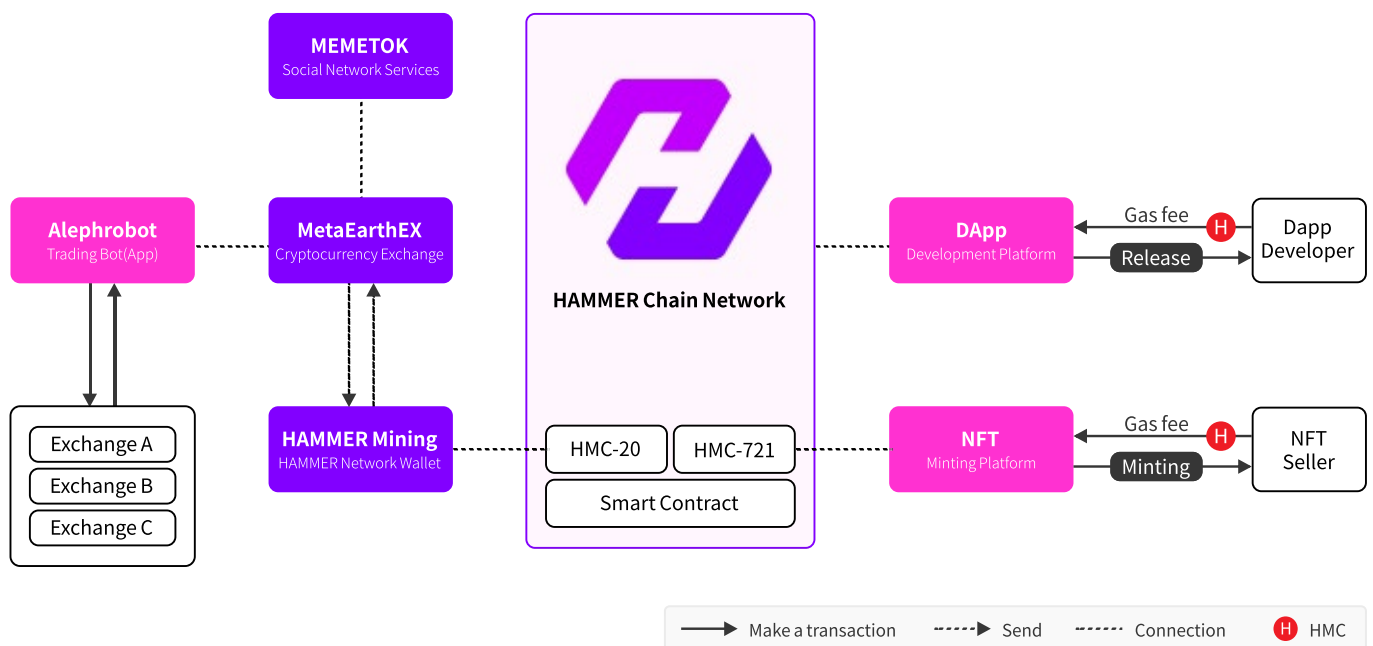


Figure08. HAMMER Ecosystem

1.1 Cryptocurrency Exchange – MetaEarthEX

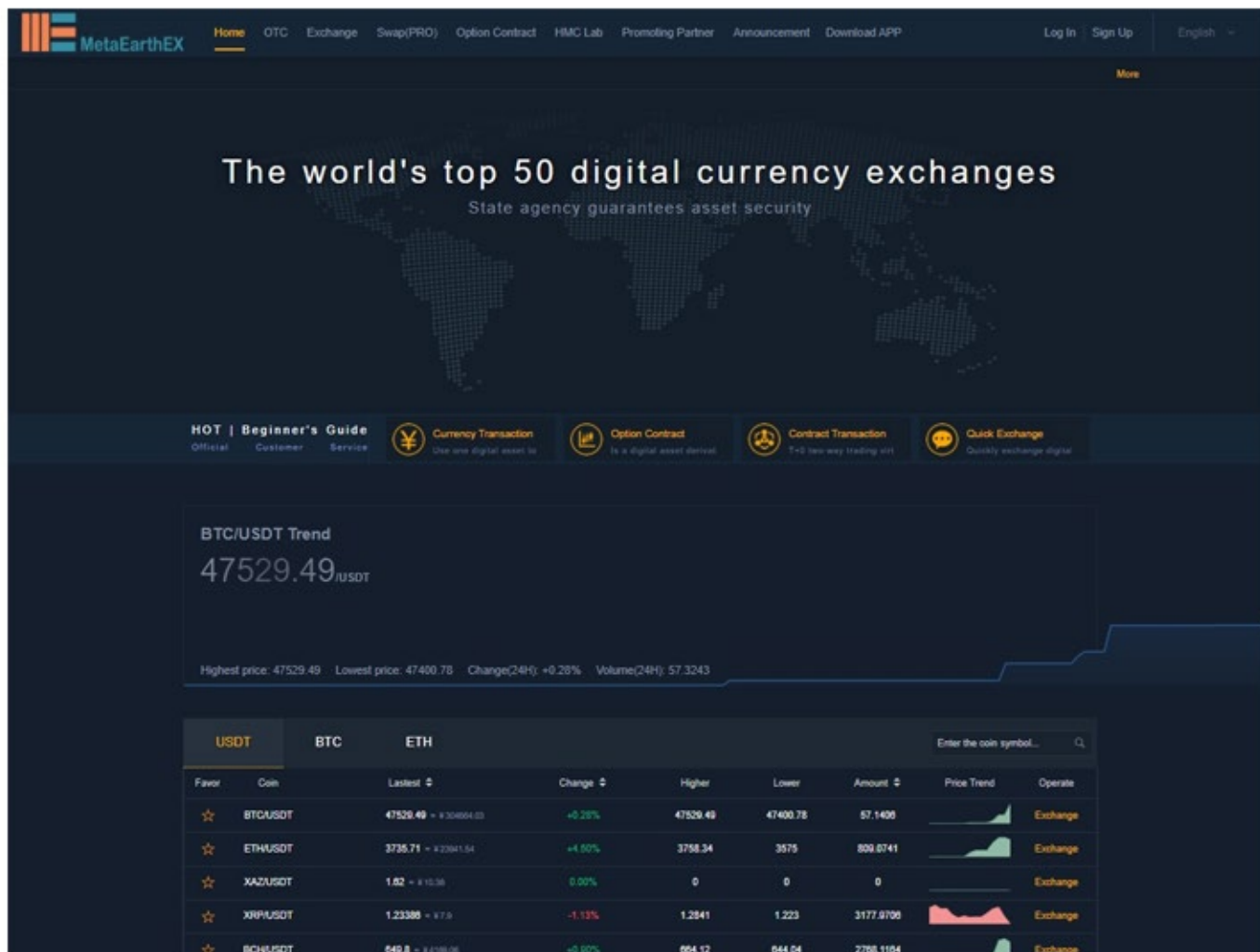


Figure09. MetaEarthEX Service Screen Capture(www.metaearthex.com)

MetaEarthEX is a global cryptocurrency exchange that supports margin trading, options trading, and NFT trading. MetaEarthEX was founded by a group of early Bitcoin participants and geeks. The core members of the team are from well-known enterprises such as Google, Microsoft, Alibaba, Tencent, and so on. It has deep research and development strength and rich experience in Internet product operation.

MetaEarthEX is located in the basic service provider of blockchain, dedicated to providing high-quality encrypted assets trading platform for global users, adhering to the basic principle of “DO NOT BE EVIL”, upholding honest, fair, and enthusiastic service to customers, and welcoming all partners/projects that are beneficial to the fundamental interests of users with an open attitude.

MetaEarthEX has the following Feature:

- Security

All-round financial wind control system and anti-theft system, hot and cold wallet, multi-signature system to ensure the safety of funds

- 24H Service

24H manual online audit can protect customers from missing out on the best investment opportunities.

- Global Exchange

Global Business Service Network Coverage Helps You Invest in Global Encrypted Assets and Transact with Global Users

- Strictly Project

We strictly select those high-quality blockchain projects to filter 80% of extremely high-risk projects for you.

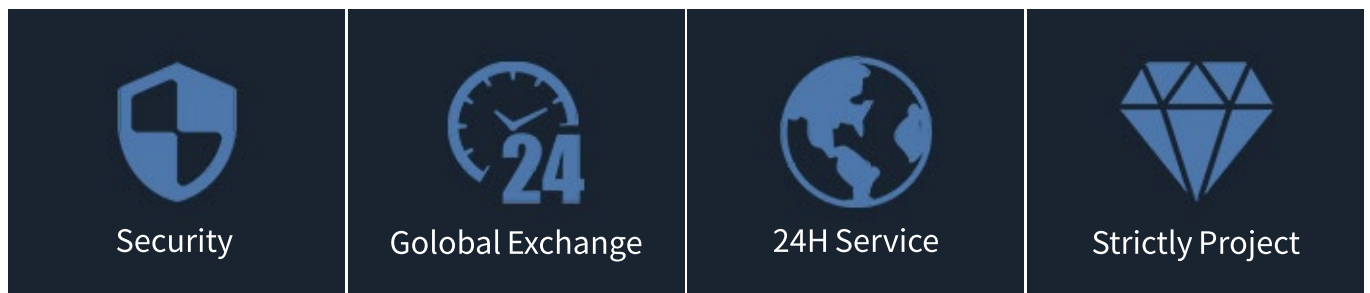


Figure10. Feature of MetaEarthEX

MetaEarthEX is a leading digital asset trading platform in the world, registered in the Cayman Islands, with a core operating team in Hong Kong. The core members of MetaEarthEX come from top Internet and financial companies. Most of them are deep believers in Bitcoin and blockchains. We are convinced that blockchains will change the traditional monopolistic financial system and build a more democratic and autonomous social structure.

MetaEarthEX platform has a professional financial-level trading architecture, a self-developed high concurrent memory matching trading engine. The platform uses a full-cold rechargeable wallet system+multiple signatures+high-protection DDOS attack system and other trading architecture to ensure customer asset security.

To let customers better grasp the investment opportunities, MetaEarthEX adopts flat management in the customer service department and establishes an extremely fast service response mechanism. The asset customer service manager is online 7*24H all-year-round, providing customers with services such as recharge and cash of assets, and ensuring that customers complete recharge in 25 minutes.

MetaEarthEX strictly screens high-quality projects and provides secure and stable asset hosting services. MetaEarthEX upholds the concept of “honesty, justice, enthusiasm, and openness”, and strives to create a safe, reliable, efficient, and friendly ultimate exchange for users.

1.2 Trading Bot(App) – alephrobot



Figure11. Alephrobot Service Screen Capture(alephrobot.com)

Alephrobot is a trading app that applied a grid trading strategy (a strategy to earn volatility profit with lower price purchase and higher price selling method). The trading bot decides the price section (range) as wide as possible and proceeds the purchase and sales at any time and operates by a method of earning the profit depending on the margin. It is known as a strategic tool where it executes the automatic purchase when the coin market price drops, and when the market price increases, it automatically concludes the sales at a high score in a price range where users decide.

Since cryptocurrency markets process transactions for 24 hours, 365 days, the volatility of the market price is very high. Therefore, by repeating the automatic purchase and sales through Alephrobot, it reduces the loss by Panic Sell (a condition of temporarily selling assets such as stocks and real estate according to changes in people's sentiment according to economic fluctuations) and Panic Buying (a phenomenon of market cornering occurs regardless of price due to psychological changes to secure as much as possible) and allows creating investing profit of users by supporting them to response the risks.

1.3 Wallet - HAMMER Mining

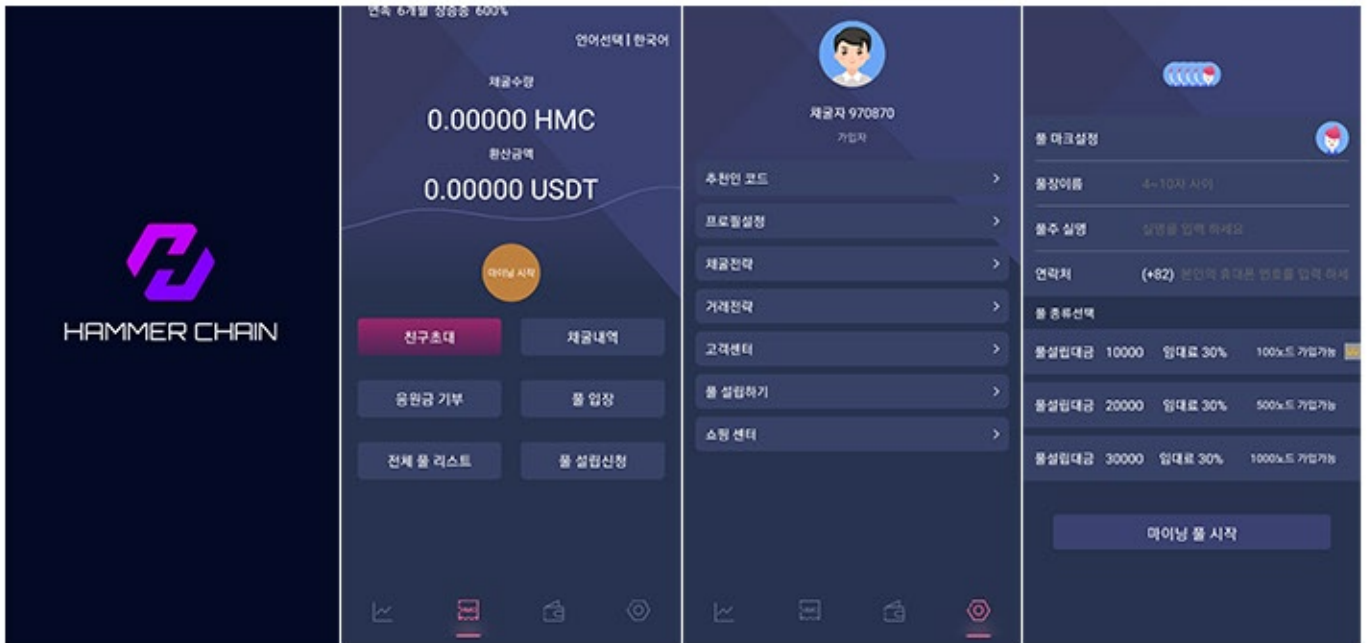


Figure12. HAMMER Mining Service Screen Capture(hammermining.com)

HAMMER Mining is an important milestone in HAMMER’s strategy of progressive decentralization towards the mainnet because every Miner will be able to interact with the HAMMER blockchain through their wallet by creating HMC transactions. Having the wallet inside the HAMMER platform will ultimately allow future HAMMER DApps and traditional businesses to easily integrate Payments and interact with the HAMMER blockchain.

To use HAMMER Mining, you need to download the app and proceed a registration. When you were invited by a friend, once your registration is done, you and your friend (inviter) will get HMC as a reward.

After signing up, users can participate by registering the already structured global pool or they can mine by establishing their pool. The batting price becomes larger as the dividend increases since the profit ratio changes depending on the fund price when the user establishes their pool.

The profit ratio of dividends depending on the funding amount is as follows :

- Funding amount 10000 → Revenue ratio 3%
- Funding amount 20000 → Revenue ratio 7%
- Funding amount 30000 → Revenue ratio 10%

1.4 SNS – MEMETOK

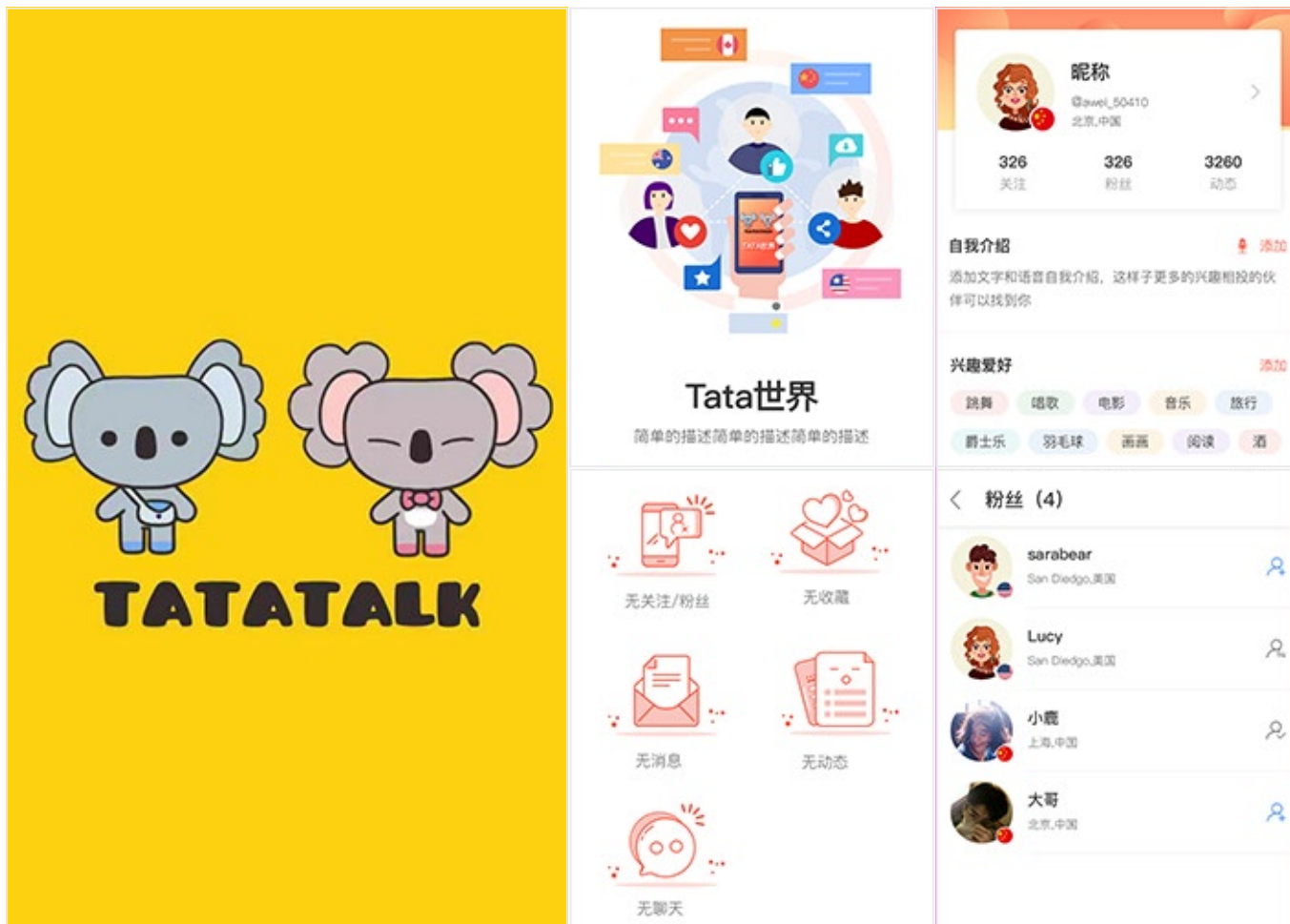


Figure13. MEMETOK Service Screen Capture(www.metaearthtalk.com)

MEMETOK is a blockchain-based new concept Social Networking Service that added a digital asset function on the basic chat function where people chat and sharing data. Digital assets can be traded on the chat room between the MEMETOK users, and it is economical as no transaction fee occurs. Also, the [Send Gift] function allows airdropping the coin to your friends.

Since it loads the function that is linked with MetaEarthEX, you can send the digital asset unlimitedly without the wallet address of the others. Anyone is allowed easy asset trading on 1:1 chat or the group chat, and it allows sending one by one in a group chat.

1.5 DApp Development

DApp(Decentralized application) is an application working on a blockchain. Like iPhone or Google makes platform with their OS and providing an ecosystem that makes app on it, the blockchain platform also allows making apps and operating them.

The perfectly decentralized application has the following characteristics.

- It is open-source and freely operated.
(It allows continuous use as it lasts forever once distributed even if the service that is operated with blockchain pauses or disappears)
- It stores data on the public blockchain.
(The data cannot be replaced or changed once it is on the blockchain)
- It gives value to the cryptocurrency.
- It generates the token based on an algorithm.

Due to the above characteristics, it is free from data falsification and applications can be operated democratically.

The developing environment for materializing DApp is divided into a backend side that writes and distributes smart contracts and a frontend side where it materializes the user interface.

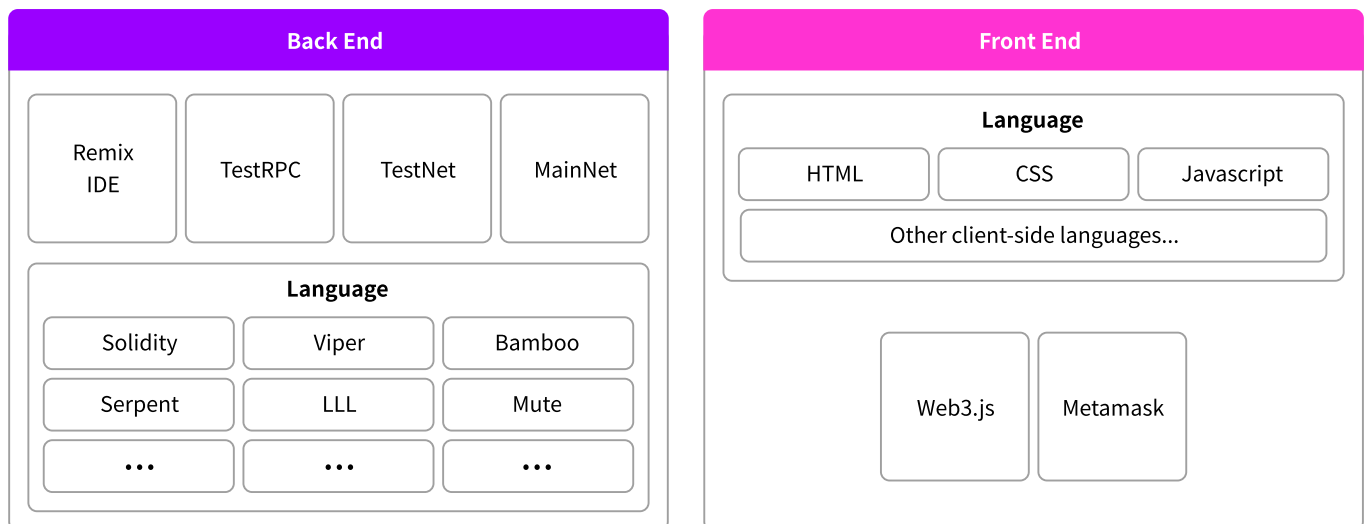
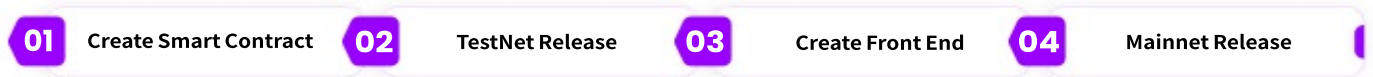


Figure14. Development environment of Back End Side and Front end side

DApp distributes contracts on the HAMMER mainnet by the following process.



Firstly, create a smart contract with Solidity, and then distribute the written Solidity code on TestNet and lastly it passes through a process of distributing the contract on HAMMER mainnet. Solidity is an objected-oriented programming language where the Ethereum based smart contract can be written.

1.6 Creating NFT

A Non-Fungible Token, NFT has a unique value therefore, it is irreplaceable with other tokens. Most of the cryptocurrencies like BTC and ETH are fungible tokens that each token has the same properties so it can be distinguished from NFT in the way these can be exchanged or traded with each other.

Through blockchain technology, NFT materializes the unique valued tangible or intangible items in the digital world and commercializes them. Since the detailed information of digital assets such as transaction history and the owners are stored on the blockchain where forgery and falsifying data is difficult, it secures the scarcity and authenticity of digital assets and can prove the ownership of the asset. It tracks and identifies digital assets and connects the users with issuers without physical limit to allow transaction and distribution.

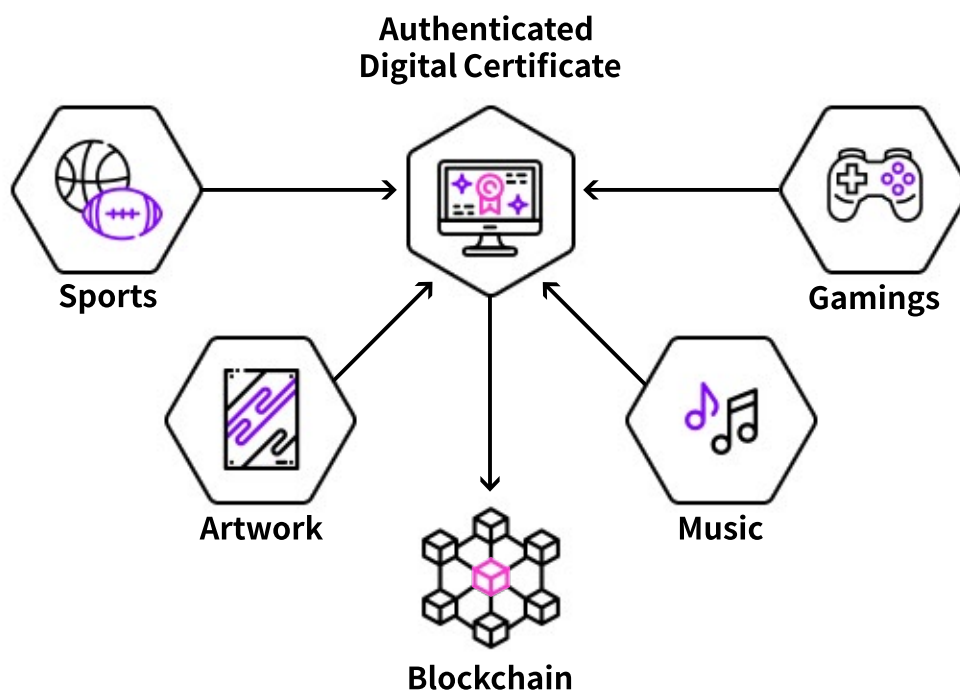


Figure15. What are NFTs

The standard protocol of NFT created by HAMMER is HMC-721. It is the expanded version of ERC-721 which is the most used NFT standard that can be exchanged with EVM. Minted with HMC-721, NFTs gives the ability to assign or claim ownership of any unique piece of digital data, trackable by using HAMMER's blockchain as a public ledger.

The minting process, from a high level, has the following steps that it goes through:

1. Creating a new block
2. Validating information
3. Recording information into the blockchain

An NFT can only have one owner at a time. Ownership is managed through the uniqueID and metadata that no other token can replicate. NFTs are minted through smart contracts that assign ownership and manage the transferability of the NFT's. When someone creates or mints an NFT, they execute code stored in smart contracts that conform to different standards, such as HMC-721. This information is added to the blockchain where the NFT is being managed.

NFT exists forever on the HAMMER blockchain once it is minted. All transactions recorded on a blockchain cannot be restored or changed. To update the NFT, tokens must be burnt and permanently deleted. During the process of NFT mint, it proceeds the code that broadcasting the digital artwork on the HAMMER blockchain in a smart contract.

NFT's have some special properties. Each token minted has a unique identifier that is directly linked to one HAMMER address. They're not directly interchangeable with other tokens 1:1. This isn't the case with NFTs. Each token has an owner and this information is easily verifiable. They live on HAMMER and can be bought and sold on any HAMMER-based NFT market. In other words, if you own an NFT you can easily prove you own it. Proving you own an NFT is very similar to proving you have HMC in your account.

The content creator's public key serves as a certificate of authenticity for that particular digital artifact. The creator's public key is essentially a permanent part of the token's history. The creator's public key can demonstrate that the token you hold was created by a particular individual, thus contributing to its market value (vs a counterfeit).

Another way to think about proving you own the NFT is by signing messages to prove you own the private key behind the address. Your private key is proof-of-ownership of the original. This tells us that the private keys behind that address control the NFT. A signed message can be used as proof that you own your private keys without revealing them to anybody and thus proving you own the NFT as well.

2. Mining Reward System

HAMMER miners receive incentives by creating blocks and transaction fees. To decide the cost of consuming computing source on the smart contract, HAMMER selected gas fee, which is the variable fee unit price model, not the fixed transaction fee. When the transaction proceeds, the gas fee is calculated according to the real computing usage amount.

To decide the cost of computing source consumed on smart contracts, the transaction fee called gas fee is supposed to be set variably when users are trading in HAMMER. In other words, when the transaction is in process, the calculated fee multiplied by the usage amount is the transaction fee of HAMMER.

HAMMER adopted the concept of Uncle Block of the Ethereum to provide a better mining environment to the miners. Uncle Block is known as the first block comprised of Orphan Blocks (When two blocks are approved at the same time, they cannot be connected to the main chain and exist separately) that passed the effectiveness of blocks but were unapproved as a final block. HAMMER gives rewards to creators and includes this hash on a network to maintain the network stably.

By adopting the Uncle Block, HAMMER allows selecting a maximum of 2 Uncle Blocks besides allowing miners to select top blocks, and rewards of each Uncle Block will be given by 1/32 of the existing block creating incentives. In addition, the main chain's Uncle Block is not treated as an actual transaction, but rewards for block generation are applied.

The mining reward of HAMMER will be as follows and the 4 types of reward will be distributed at the mining pool hub.

- 1) Block creation reward : 2 HMC
- 2) Transaction fee of used up gas
- 3) Uncle reward : 1.75 HMC, 1.5 HMC, 1.25 HMC, 1 HMC, 0.75 HMC, 0.5 HMC
- 4) Uncle included incentive : 0.125 HMC, 0.0625 HMC

3. Roadmap



• 2021

- Presenting prototype
- Hammer API development
- Mainnet setup
- Presenting stable asset related standard
- Establishing Social Network Wallet Services plan



• 2022 1Q

- Creating testnet contract
- System development related to staking
- Preparing tutorial data
- Completing contract and announcing open source
- Creating private asset on testnet and complete contract
- Finalise API function
- Presenting Social Network Wallet Services 1.0 Version



• 2022 2Q

- Creating mainnet contract for delegation
- Creating staking mandate reward on mainnet
- Optimizing gas
- Supporting wide range service through testnet upgrade
- Expanding Social Network Wallet Services



• 2022 3Q~4Q

- Expanding existing infra
- Optimizing Social Network Wallet Services
- Supporting variety of private asset on mainnet
- Announcing documents such as development guidebook
- Developing bigdata based search

CHAPTER 4

Legal Disclaimer

This whitepaper briefly summarizes the core concept of HMC This whitepaper was written for the purpose of providing necessary information to those who want to participate in the project, and is not a document that recommends investment, etc.

1. General Notice

The project-related coins in this whitepaper do not correspond to financial investment products such as securities, and their issuance does not correspond to subscription for financial investment products or solicitation of subscription. HMC is not intended for speculation and have no rights in any form of real estate, intellectual property rights, other property, or cash.

HMC is not stocks because they do not give ownership to the distributor, and ownership of HMC does not give the right to participate in the decision-making process in assets and/or business plans. In addition to the benefits that can be obtained from the use of the platform, there is no commitment to the value or rights for the separate HMC profit.

The information contained in this whitepaper was prepared on the date indicated on the cover. Including information on the business operation and financial condition of the distributor in the future, information written in this whitepaper may be changed from time to time.

The provision of this whitepaper and sales at the project stage do not state that no changes have been made to the content of the whitepaper. Distributors do not make or claim in any form (both explicit and implicit) that the information contained in this whitepaper is up-to-date, accurate and complete, and expressly deny it.

This whitepaper may contain information from third parties and/or the content of publications. All information and data reproduced in this whitepaper are deemed to have been derived from reliable sources, but distributors do not separately verify such information or data and give no warranty as to the accuracy or completeness of such information or data.

Neither the full text nor partial excerpts of this whitepaper may be regarded as advice on legal, financial, tax, or other professional provisions. You should get separate professional advice in your decision to purchase HMC Please note that you are solely responsible for all evaluations and decisions that may arise when deciding whether to purchase HMC You may request additional information regarding HMC to your distributor.

Distributors have discretion, but not legal force, and provide the requested additional information in consideration of all relevant circumstances, including whether the request for information is reasonably necessary such as ① whether there are any prohibitions or restrictions from a legal point of view, ② whether the requested information is a matter of helping the reader to further clarify what is specified in this whitepaper, and to seek expert's advice, whether it is something that could influence his/her/its decision as to whether the reader buys coins.

Please note that this whitepaper was prepared for the purpose of information delivery and has not described the future prospects. It is stated that distributors are responsible for all losses and damages (direct/indirect, predictable, or others) arising from acting or relying on any information related to the distributor or the contents of the HMC ecosystem contained in this whitepaper or additional information inquired by another reader even if there was the distributor's negligence, default, or insufficient guidance.

Reproduction of this whitepaper except for this warning and notice is strictly prohibited and is not permitted by distributors. The information set out below may not be complete. No part of this whitepaper forms a legal relationship between the recipient of this whitepaper and the distributor, and it is structured so that the recipient of this whitepaper cannot be legally binding on the distributor nor enforce any legal action.

In purchasing HMC, you are deemed to have thoroughly reviewed the whitepaper, and also deemed to have agreed to the terms and conditions of the provision of HMC. Please note that the information specified in this whitepaper does not guarantee any legal liability. You expressly acknowledge and agree that HMC are not securities and do not generate any form of return on investment.

HMC and related services have been designed to be used for the purposes specified in the whitepaper and are provided by the distributors as they are and under this use if possible. Distributors do not make any guarantees or representations (both explicit and implicit) of the accessibility, quality, suitability, accuracy, adequacy or completeness of HMC and related services, and expressly deny this.

Thus, the distributors also expressly deny any liability arising from errors, delays or omissions in any action taken by you by relying on HMC and related services. No warranty is made in any respect, including the guarantee that there will be no infringement of the rights of third parties with respect to the HMC and related services provided by the distributor, name, merchantability, satisfactory quality, or fitness for a particular purpose.

We cannot guarantee the future performance and value of HMC. This includes the intrinsic value of HMC, and also includes a guarantee that it will have any value in the future. Unless you fully understand and accept the distributor's overall business plan and the potential risks of HMC, you should not participate in the sale of HMC.

HMC was developed for the future function of the HMC ecosystem. The distributor expressly denies full responsibility for any losses you may suffer in connection with the purchase of HMC.

You are not obligated to make any contract or binding legal commitments in connection with the sale or purchase of HMC. A separate document describing the terms of the contract (hereinafter "terms of the contract") is applied to the contract between you as the distributor and the purchaser and to the sale and purchase of HMC. In the event of inconsistency between the terms of the contract and the contents of this whitepaper, the former takes precedence.

2. Risk factors

- Purchase of tokens

HMC should be purchased only by a financial expert who fully recognizes and evaluates the advantages and risks of the purchase, or a person who can get advice from a token trading-related professional advisor, or a person who can withstand the losses that may be incurred by purchasing, including the possibility of losing the total amount spent on the purchase of HMC.

- No prior market

No guarantee is made as to whether the HMC-enabled market or liquid funds trading market will be developed, and it must be sustainable as HMC are traded on the cryptocurrency exchange after the development is completed.

- Platforms that have not been completely developed

The value and demand for HMC is highly dependent on the performance and commercial success of the HMC platform/service. Distributors give no warranty as to whether the HMC platform/services will be commercially successful. In addition, the HMC platform/service has not been fully developed, finalized, and integrated, and additional changes, modifications, updates, and adjustments may exist before release (and after release). These changes can cause unpredictable consequences to users, which in turn may affect success.

- Risks related to uncertain losses

HMC is not protected unless personal insurance is separately acquired as a means of protection for HMC. Therefore, there will be no separate public or private insurance as action taken by the distributor for the buyer in case of loss of the token itself or the utility value of the token.

- Tax-related risks

The tax characteristics of HMC are unclear. Therefore, the tax system that will target the token is also uncertain. Anyone who wants to buy HMC should find a personal accountant who can deal with related issues before deciding whether to purchase HMC. Distributors do not make any statements as to whether taxes may be incurred by purchasing or holding HMC.

- Risks from HAMMER Protocol

Due to the nature of HMC and HMC platform/service based on HAMMER Protocol, malfunction, breakdown, suspension, or disposal of HAMMER Protocol may have a significant negative impact on HMC and HMC platform/services. Advances in crypto technology or advancement of related technologies, such as the development of quantum computing, may be a potential risk for HMC and HMC platforms/services. This includes the use of tokens to acquire services, rendering inefficiency of the password matching mechanism that supports the HAMMER Protocol, and other risk factors.

- Risk from third parties

HMC is an asset based on the blockchain technology. The security, mobility, storage, and connectivity of these blockchain assets are based on factors such as the security, stability and suitability of the basic blockchain protocol and process (In this case, HAMMER is out of the distributor's control.)

Unexpected events may occur, such as mining attacks, hacking, and unauthorized access to the private key of the wallet where HMC are stored. Distributors cannot guarantee that they will prevent these external factors from adversely affecting HMC directly or indirectly.

Furthermore, the risks posed by third parties also include illegal acts, fraud, and failure to receive HMC at the time of token payment because the third party's wallet is not compatible with HMC. Any resulting losses are irreversible. Distributors are not responsible for any risks arising from the intervention of third parties and cannot take action to recover lost HMC in this way.

3. Anti-Money Laundering / Combating the Financing of Terrorism

As part of the token distributor's responsibility in relation to the policy for Anti-Money Laundering / Combating the Financing of Terrorism (“AML-CFT”), detailed identification of anyone wishing to own or use HMC as a means of payment is required. Depending on the circumstances of each application, token distributors may request additional information and/or documents to the applicant from time to time.

Pursuant to all applicable laws, distributors are committed to complying with all relevant guidelines applicable to their business with respect to AML-CFT obligations. Distributors reserve all rights to exclude anyone who refuses to provide appropriate information and/or documents that may be requested by the distributor from the purchase process in the process of initial coin purchase and/or further purchase of HMC

In addition, distributors reserve the right to prohibit the handling or provision of financial services to designated individuals and legal entities for any suspicious transaction. In this case, distributors have all rights to refuse any service/ HMC transactions.