# VIV White Paper

2022.06

# Abstract

The internet is iterating towards web3. web3 is changing the technical architecture and business architecture compared to the traditional internet, smart contracts have become the new infrastructure, all web3 products including defi, nft, DAO, are based on smart contracts to operate. In the web3 era, users, on-chain data, off-chain data, digital assets, are interacting around smart contracts. The future re-implementation of many offline scenarios and traditional Internet scenarios in web3, as well as the development of new scenarios, will give rise to a massive demand for smart contracts. At the same time, the design, development and use of smart contracts have a high threshold, which has become a huge obstacle to the development and popularity of web3.
To solve this problem, VIV proposes the concept of "payment protection by smart contracts" to help inexperienced users access web3 with one click, allowing them to manage their own assets and complete transactions with other users using smart contracts without programming knowledge. At the same time, using the characteristics of smart contracts can not be tampered with, to ensure the normal conduct of transactions, prevent losses caused by malicious behavior, and protect the security of crypto assets.

# Introduce

## background

Starting with the birth of Bitcoin in 2008, developers worldwide set out to build a decentralized network of value. As of 2022, the web3 world has formed a developed ecology. But at the same time, the total number of web3 users is still relatively small and there is still a lot of room for growth. Data shows that metamask has about 30 million monthly active users and the active address of the ethereum network is about 6 million, compared to Facebook, which currently has 3 billion monthly active users and tiktok, which has 1.2 billion monthly active users. We believe that in the coming period, web3 will penetrate from the enthusiast community to a broader group of users. At this stage, the application scenario of web3 will be greatly expanded.
The current web3 ecosystem only covers a limited number of scenarios, such as going Defi, NFT, DAO, Gamefi, etc. According to the Federal Reserve's survey on the financial status of the US population: 12% of people hold various types of crypto assets in 2021,

11% of them as investments, and only 1-2% of them for shopping or money transfer, which indicates that there is still a large amount of online and offline economic activities that have not been entered into web3. in 2011, Internet practitioners proposed Why Software Is Eating In the following decade, almost every traditional industry and government organization has undergone an Internet and software-based re-architecture. We believe that with the gradual penetration of web3 into the traditional world, industries will need to re-build their transaction processes based on web3. At the same time, with the development of new technologies such as the metaverse, there will be economic models that people never imagined would continue to emerge.

## Technical requirements for smart contracts

There are huge differences in the technical architecture of web3 compared to web2.

| | data layer | logic layer |
|---|---|---|
| web2 | databases | server |
| web3 | Blockchain + Database | smart contract |

In this architecture, smart contracts play a central role. Users, on-chain data, and off-chain data all need to interact with smart contracts and ultimately settle digital assets through them.
Therefore, smart contracts play a crucial role in the technical architecture of web3. With the advent of the web3 era, the world is about to experience an explosive growth in the demand for smart contracts.

# Issues

## Flexibility of contracts

Many of the current web3 products are built on a limited number of smart contracts, with the contracts maintained by a team of professionals. But this format will not meet people's future needs. For example, I need to enter into a short-term lease for a parking space with a community that contains more complex rights and responsibilities clauses, which would need to be written into the contract at the editing stage. The reality is that a huge number of transactions occur every day, each one different from the next, and some of them may only need to be done once, which places a higher demand on the flexibility of the contract.

## Development threshold

There is a high technical barrier to the development of smart contracts. Developing a usable smart contract requires the developer to have a deep understanding of the business logic, proficiency in the contract language, and the development of a user interface that can interact with the contract. Also, there is a need to ensure that the contract is free of security risks to prevent exploitation by attackers; and the contract needs to run efficiently to save on miner fees.
The technology in the web3 world is still rapidly iterating, with new public chains and new protocols appearing all the time, requiring developers to keep up with the technology. As a result, smart contracts place higher demands on developers compared to software development in web2.

# Project definition

Based on the above problems, we believe that the technical threshold and time cost of smart contract development need to be significantly reduced, so that ordinary people with no programming knowledge can develop and deploy a smart contract in minutes. We define this goal as a "payment protection by smart contracts".

# Technology implementation

## Contracts Code Base

VIV builds a modular contract code base in which functional modules commonly used by users are independently encapsulated. The code base serves as the underlying resource for the project to call, and the code modules generate complete smart contracts based on business logic in a process similar to building blocks. The code modules in the project cover multiple public chains and will be continuously updated and iterated.
Cover more public chains: will support public chains with greater user demand according to the ecological development of each public chain
Support more functions: As the web3 ecosystem continues to expand, the types of business, protocols, and digital assets that users need will gradually increase, and the code base will continue to increase the number of functional modules.
Code optimization: existing code modules, optimizing the code towards low cost and operational safety.

## user interface

We have built a cross-platform user interface, including web side, mobile H5 side, iOS, Android side. The user interface contains two major functions, wallet and transaction,

where the user can achieve the following operations in the transaction function.
1. Submitting requirements and initiating contracts
2. Interacting with already deployed contracts
The specific interactions are described below.

## Deployment platform

We have built the deployment platform in the back-end server. The deployment platform contains the following components.
1. The contract engine, which is responsible for automatically generating business processes based on user instructions, and calling relevant code modules from the contract code base to compose complete contracts and deploy them to the public chain.
2. Page engine that generates front-end pages for interacting with smart contracts based on the user-generated smart contracts, as well as with the corresponding transaction process, permission information, etc.

## Databases

The characteristics of in-chain stored data are high cost and permanent storage. In real business, much of the data is not necessary for permanent storage, and to save costs, this data will be stored in the back-end database. In future versions, users will be allowed to choose their own way of storing data and pay for it accordingly.

# Project route planning

In order to achieve the above vision, we have divided the project into three phases.

## Phase I Tools

1. Support for smart contract-based online trading
2. Covers a wide range of transaction scenarios
3. Covering BTC, ETH, BSC, Tron networks
4. Development of mobile portal

## Phase II Platform development

1. Establishment of a trading plaza
2. Support for publishing goods tied to smart contracts
3. Provide transaction interface for b-end users

## Phase III

1. Establishment of community governance mechanisms
2. Building an economic system based on community ecology

# Project components

In the roadmap, the first phase of functionality has been largely developed and includes the following main features.

## Trading Model

In the first version, we implemented the following 8 transaction modes in the form of pre-built transaction modes. Each of these 8 transaction modes provides configurable parameters that can be used to generate the corresponding smart contract and deploy it to the chain based on the parameters provided by the user.

### General trading

One to one single transaction for physical, digital, domain, NFT, fiat, etc.

### Auction transactions

The buyers will bid against each other and the highest bidder will win.

### Split transactions

Buyer pays once, seller withdraws multiple times. Can be used for transactions that include warranty, deposit, and other terms to help build long-term trust between the parties.

### Regular trading

Transactions that require regular payments in the future, such as rentals and labor partnerships, can use deposits to build trust.

### Trust fund

Entrust digital assets to a smart contract for management and designate beneficiaries, and the contract will automatically dispose of the assets according to pre-defined rules.

### NFT mortgage lending

Post lending needs by holding NFTs as collateral.

### DAO

For the management of funds in decentralized autonomous organizations (DAO).

### Project Crowdfunding

Post a crowdfunding campaign to raise funds in one go.
We will be developing more trading models in the future, as well as increasing the number of configurable options in our trading models to provide greater flexibility for users to meet a wider range of needs.

## Transactional arbitration

According to incomplete contract theory, one cannot anticipate all possible scenarios when drawing up a contract, hence the incompleteness of real-world contracts. Similarly, for smart contracts that map to real-world transactions, it is impossible to cover all possible future events when the contract is drawn up. Therefore, there needs to be a reserved interface in the contract to deal with the occurrence of unexpected events. Based on the above considerations, we provide a transaction arbitration function in the contract. When initiating a transaction, the initiating party can choose the arbitration method; when a dispute arises in the transaction, it will proceed to arbitration according to the pre-defined rules. The arbitrator will make an arbitration based on the evidence uploaded by both parties to arrive at the final attribution of the assets in the smart contract.
There are currently three types of arbitration.

### Platform arbitration

Arbitration by VIV project parties

### Third-party arbitration

The transaction initiator designates a mutually recognized public chain address as the arbiter, such as a credible individual, institution, judicial department, etc.

### Community voting (in development)

A community vote to determine the final ownership of assets in the contract
After reaching the arbitration conclusion, the arbitration party will sign the arbitration conclusion and submit it to the smart contract, which will distribute the digital assets based on the conclusion. At the same time, a portion of the assets will be allocated to the arbitration party as a reward for the act of arbitration.

## Wallet function

In order to facilitate user interaction with smart contracts, we have developed a wallet function that supports multiple public chains and supports the following features in addition to basic wallet functions such as transferring funds and easily interacting with contracts:

### Multi-signature wallet

Used for multiple people to manage a sum of money and only after confirmation from a specified number of people can the transfer be completed.

### Secure transfers

A one-to-one transfer transaction can only be completed once the transfer is confirmed by the recipient.

### Fund Pooling

Funds from multiple wallet addresses can be pooled into a single address.

### Batch transfers

Allows simultaneous transfer of funds to a range of recipients

# Economic mechanisms

## Source of income

In the first phase of the project, the income of the project owner consists of the following entries.
1. Platform service fee. The project owner charges a service fee to the recipient of each transaction.
2. Arbitration fees. Arbitration revenue is earned in arbitrations in which the platform party is involved.
In the second and third phases of the project, fee rules will be set for new features.

## Economic system

We are committed to making VIV an active community with multiple participants and will give back to our community users in many ways.
In the first phase of the program, users can invite friends to sign up to receive complimentary coupons for reduced fees for the platform's services, and can participate

in arbitration to receive arbitration fees.
In the second and third phases of the project, a more sophisticated economic system will be designed to give all platform participants the opportunity to participate in the platform and reap the rewards.

# References


1. The Ownership Economy 2022 https://variant.fund/writing/the-ownership-economy-2022
2. Why Software Is Eating the World https://a16z.com/2011/08/20/why-software-is-eating-the-world/
3. Incomplete contracts theory
https://en.wikipedia.org/wiki/Incomplete_contracts#Limitations
4. Economic Well-Being ofU.S. Households in 2021 ( page 45) https://www.federalreserve.gov/publications/files/2021-report-economic-well-being-us-households-202205.pdf