Smart Contract and Token System

by Sunny King Team

March 31, 2019

Since the creation of blockchain, there are numerous updates and changes made to the technology. However, with "open source" being a commonplace in the blockchain industry, thresholds and costs for clone replication remain very low. Most of the many iterations of blockchain are nothing but a small change on the terminology.

On another hand, the discrepancy between an innovative programmer and an outstanding business leader often leads to a gap between the technology and actual adoption. The expectation on short term profit also plays a part in hindering the development of a blockchain project, as the balance between investors, community, and users becomes a constant struggle.

To rectify all the issues that prevent a project from reaching maturation, we at v.systems set a steady and gradual development path. From our concept inception in 2016 to our main chain launch in November 2018, our next step in realizing the vision of our project is to release the first version of smart contract function in the Q2 2019, followed by an update in the core of database applications (data storage management) later this year.

1. Smart contract

Nick Szabo's paper "Formalizing and Securing Relationships on Public Networks" was the first comprehensive documentation of smart contracts. According to his definition, a smart contract is a digital protocol for data transmission. It uses cryptographic algorithms to automatically execute transactions and complete the control processes once the given conditions are met.

With the birth of Bitcoin, the smart contract technology has since evolved, allowing for trusted transactions without the involvement of third parties, under which the transaction is transparent and can be tracked, but not tampered.

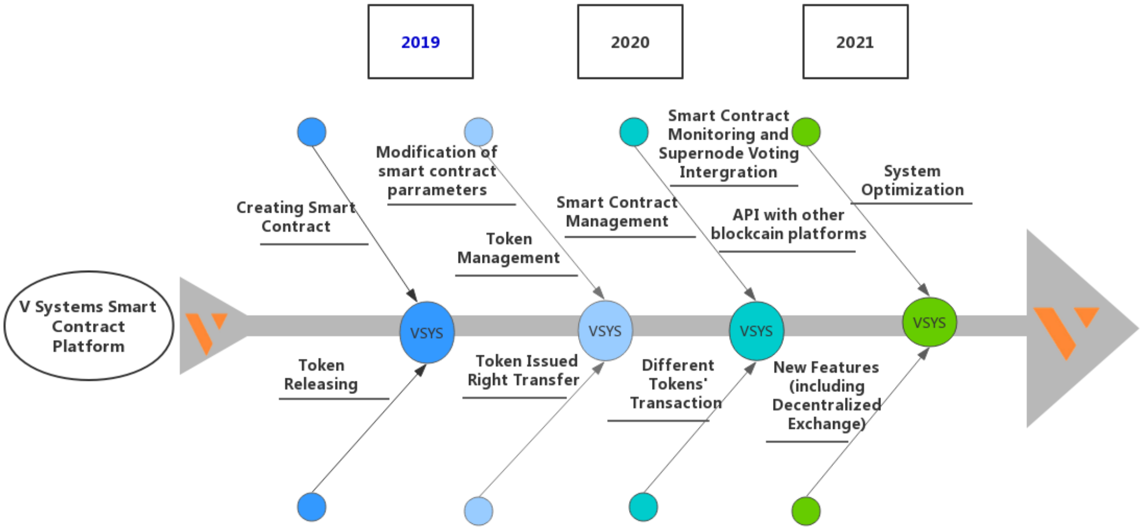
Currently, there are two directions for smart contract technology development— non-Turing-complete and Turing-complete. Bitcoin belongs to the first category and carries a non-Turing-complete scripting language. Custom smart contracts can be created on Bitcoin, such as multi-signature, hosting, time locking and many more.

Ethereum, on the other hand, belongs to the second category and uses the new Turing-complete language Solidity. Through the Ethereum platform, smart contracts with wider diversity and more complex terms can be developed. The downside of this is that complex smart contract makes it more difficult to analyze and thus increases the likelihood of finding vulnerabilities.

Common advantages of smart contracts on blockchain:

- Higher efficiency: business processes and manual processing time are reduced by code execution
- More autonomy: no intermediary is required, and execution is an automatic process when consensus is reached
- Higher accuracy: reduced errors in operation
- Lower cost: massive time savings and reduced overhead operating costs
- Better security: encryption mechanisms and multiple signatures ensure the security of files and data

Considering the technology development and industrial needs for smart contracts, v.systems will temporarily adopt the non-Turing-complete scripting language, so that smart contracts can be secure, resource-efficient, and easy to use and manage. In the near future, a Turing-complete model will eventually be adopted by v.systems.


2019
2020
2021
V Systems Smart Contract Platform
Creating Smart Contract
Modification of smart contract parrameters
Token Management
Smart Contract Management
Smart Contract Monitoring and Supernode Voting Intergration
API with other blockcain platforms
System Optimization
VSYS
VSYS
VSYS
VSYS
Token Releasing
Token Issued Right Transfer
Different Tokens' Transaction
New Features (including Decentralized Exchange)

Note:

1. Smart contract ownership cannot be transferred, but the token issue right can be transferred. The contract creator has the final right to interpret the token issue right.

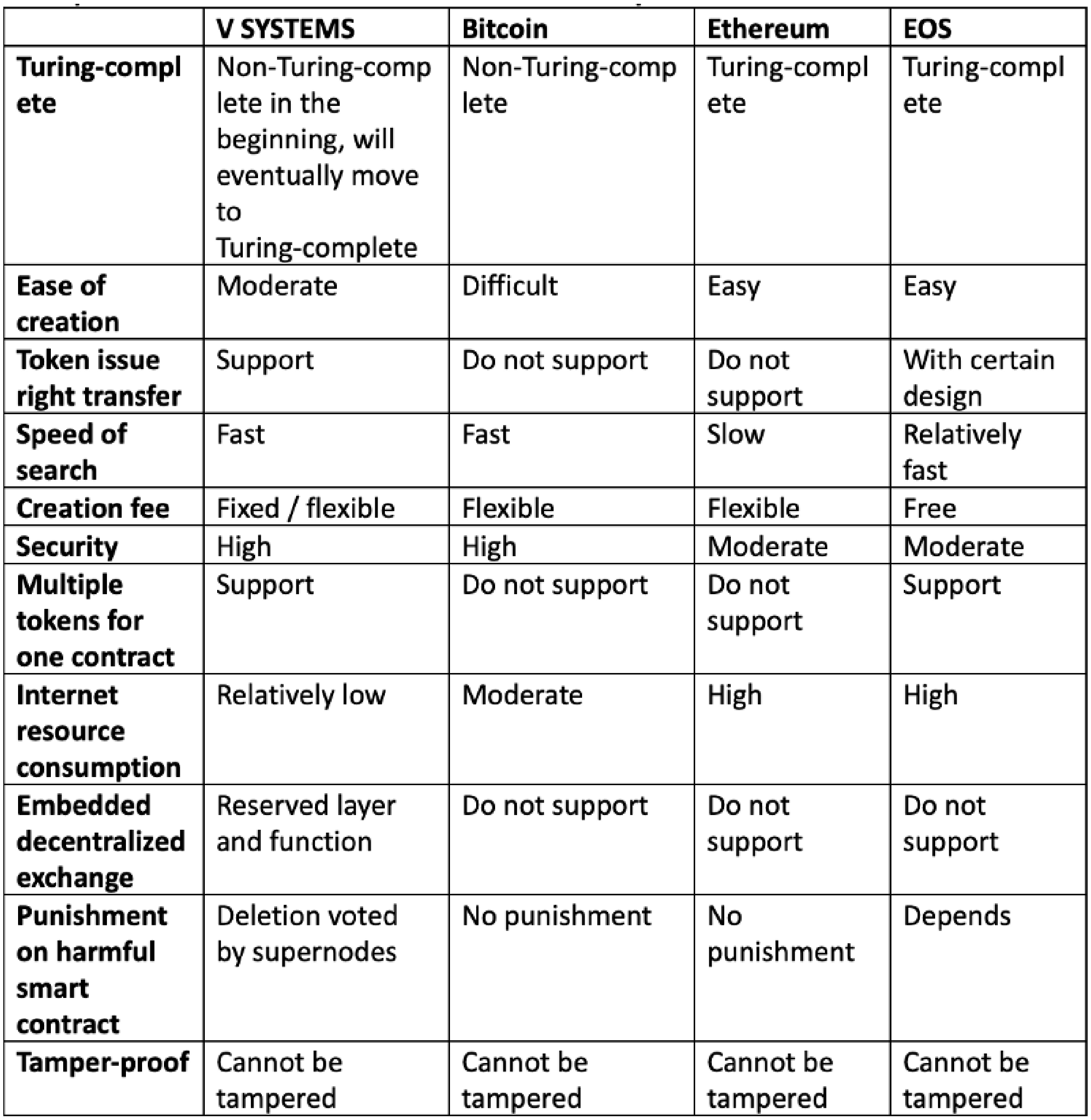
2. The smart contract itself cannot be modified. It is a simple consensus and cannot be modified at will, but the parameters of some contracts can be changed. The contracts with modifiable parameters are relatively weak in consensus. These parameter revisions will provide choices and an advanced notice.

v.systems’ smart contract development will be separated into three phases. The first phase is the initial simple creation and delivery of the token publish function. The second phase emphasizes trading and management, and the third phase targets to optimize performance.

Function details:

- Smart contract creation
- Token publish (token destruction and issuance)
- Contract parameter modification
- Token issue rights transfer (contract creator has the final decision)
- Smart contract management

Comparison between smart contracts on various platforms



	V SYSTEMS	Bitcoin	Ethereum	EOS
Turing-complete	Non-Turing-complete in the beginning, will eventually move to Turing-complete	Non-Turing-complete	Turing-complete	Turing-complete
Ease of creation	Moderate	Difficult	Easy	Easy
Token issue right transfer	Support	Do not support	Do not support	With certain design
Speed of search	Fast	Fast	Slow	Relatively fast
Creation fee	Fixed / flexible	Flexible	Flexible	Free
Security	High	High	Moderate	Moderate
Multiple tokens for one contract	Support	Do not support	Do not support	Support
Internet resource consumption	Relatively low	Moderate	High	High
Embedded decentralized exchange	Reserved layer and function	Do not support	Do not support	Do not support
Punishment on harmful smart contract	Deletion voted by supernodes	No punishment	No punishment	Depends
Tamper-proof	Cannot be tampered	Cannot be tampered	Cannot be tampered	Cannot be tampered

Future prospect of smart contract

Gavin Wood, creator of Ethereum smart contract codes, once said that smart contracts carry great potential to change the society and will serve as a basis for those changes.

The first areas to be considered for v.systems smart contracts are:

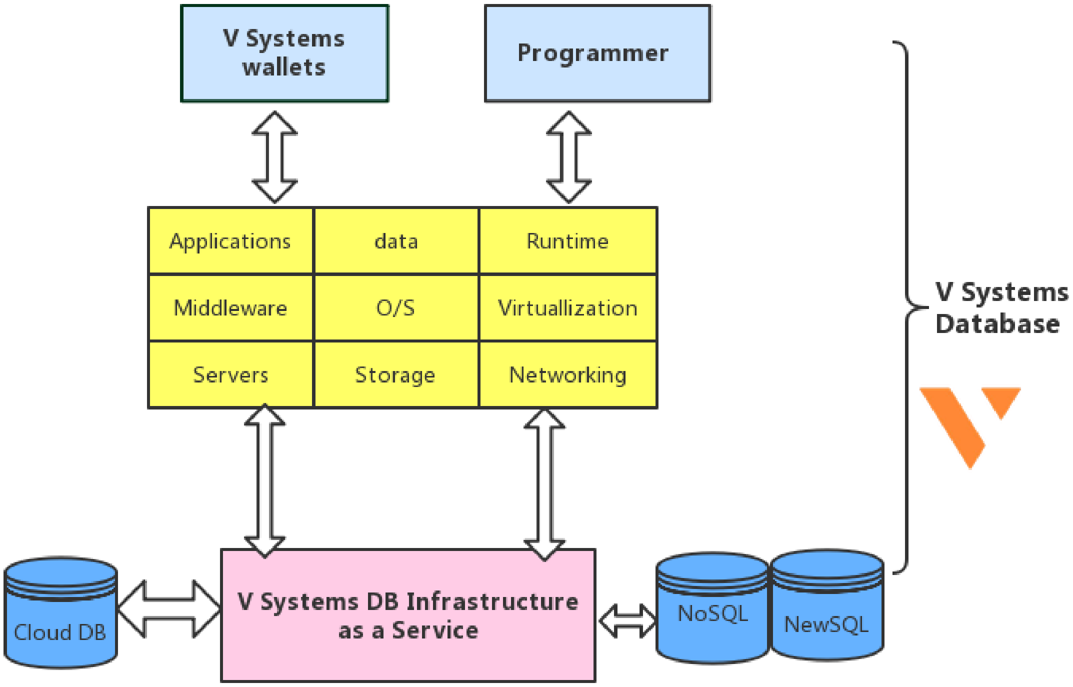
i. Asset digitization
ii. Project financing
iii. Insurance
iv. Health
v. Personalized data
vi. Advertising
vii. Gaming

2. Database storage management and billing

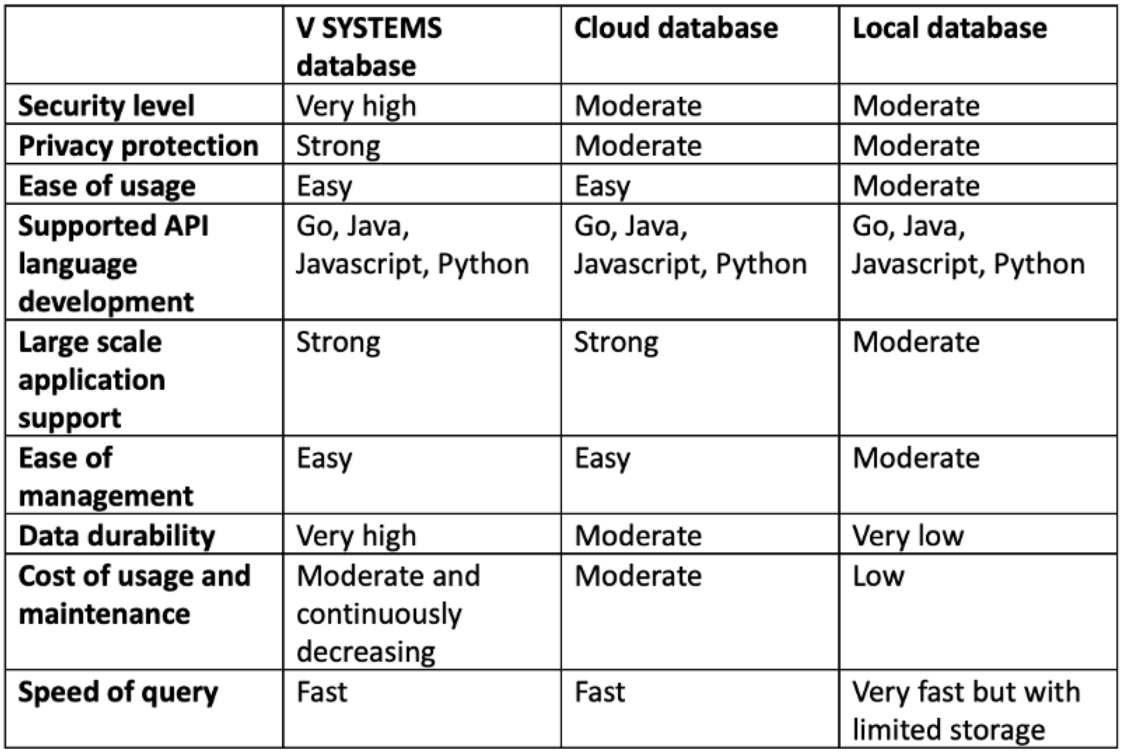
v.systems has always advocated for a reorganization of blockchain database function to link the characteristics of blockchain (duplicating and immutable) to general database. This could avoid the cumbersome and difficult management of traditional database usage.

In v.systems's public chain design, there are two initial steps to materialize the industrialization of blockchain database:

- The first step is to achieve secure storage management and data billing, which is one of the basic functions of the database.
- The second step is to gradually improve other functions, including seamless integration with traditional centralized databases, database performance enhancements, optimization and fast index compression.



The design will be a cloud blockchain database architecture. Developers will not feel the difference in the backend and can easily use the database through v.systems through only managing public and private key pairs on v.systems' wallet.
Compared to traditional database, v.systems' blockchain database carries unique advantages:



	V SYSTEMS database	Cloud database	Local database
Security level	Very high	Moderate	Moderate
Privacy protection	Strong	Moderate	Moderate
Ease of usage	Easy	Easy	Moderate
Supported API language development	Go, Java, Javascript, Python	Go, Java, Javascript, Python	Go, Java, Javascript, Python
Large scale application support	Strong	Strong	Moderate
Ease of management	Easy	Easy	Moderate
Data durability	Very high	Moderate	Very low
Cost of usage and maintenance	Moderate and continuously decreasing	Moderate	Low
Speed of query	Fast	Fast	Very fast but with limited storage

With v.systems database solution, privacy leaks like in Facebook could be completely avoided, and the protection of personalized data will be greatly enhanced. Game player’s ID and virtual assets will be permanent, and no one can deprive him of handling the data. Shortcomings brought about by centralized applications can be gradually eliminated.

v.systems will continue to evolve in accordance with the established development roadmap in 2019. All function updates will be designed for industrialization.

Exciting updates are just around the corner.