FEVERTOKENS

# UNLOCK SMART CONTRACT SCALABILITY

Engineering an ecosystem of composable modular smart contract packages

DEC 2023

https://fevertokens.io

Introducing

# Functional Scalability in Web3 Applications

Web3 has the potential to lead once-in-a-lifetime paradigm shift in how technology, markets, and capital interact. At the present time, however, this potential remains untapped, hindered largely by the functional scalability challenge. Here, we explain what functional scalability is, why it can unleash the market potential of Web3 and blockchain, and how we can build it.

While still in its infancy, Web3 has quickly grasped the attention of institutional players who had played catch-up in the blockchain space until recently. For example, keen observers from Citi Group project the value of tokenized financial and real-world assets to be over $4 trillion by 2030. Similar optimism is found in reports from McKinsey, BCG, and Goldman Sachs, among others.

The main appeal of Web3 resides in the range of qualities that its underlying technology enable. Web3 enhances trust and transparency, which have broad applications in supply chain and finance. It allows apps to incorporate more personalized experiences with ease, while preserving the privacy of the user. Decentralized networks on which the core Web3 architecture operates is resilient to failures and attacks. Meanwhile, with global connectivity and reduced number of intermediaries, the efficiency of economic activities mediated by Web3 is higher...

Advantages like these have excited corporations and financial institutions to explore the Web3 space. However, most still struggle to pull together business-ready projects and drive value from this endeavor. Our works with larger players like this have demonstrated clearly that existing approaches to build Web3 apps are incongruent with the need of corporations for high flexibility, modularity, interoperability, composability, security, reliability, governance, and compliance directly integrated into the design of smart contracts and their underlying architectures.

**In short, the need for functional scalability is not being met.**

## About FeverTokens

FeverTokens positions itself to be the leading **fabric of advanced, application-level blockchain protocols, especially for real-world asset tokenization**. Our protocol builder already underpins highly-scalable and sophisticated tokenization projects. We have a unique deeptech approach that enables both functional scalability and enterprise-grade, built-in security with formal verification.

How can we build functional scalability?

# Package-Oriented Web3 Framework

Many aspects of functional scalability must be built directly into each of the smart contracts from the ground up. One may also hope for a tool or plug-in that can be added to achieve functional scalability. However, the nature of blockchain means that functional scalability must be native to the Web3 app.

While it may also be tempting to write “quick and dirty” codes to validate the market, the lack of functional scalability can quickly punish this approach as the cost of market validation piles up. Hence, functional scalability is best taken into account from the very beginning.

With these considerations in mind, we have devised the industry’s first package-oriented Web3 framework that empowers builders with fast and low-cost tools to build functionally scalable Web3 apps.

## Why is functional scalability important?

Encompassing flexibility, modularity, interoperability, composability, reliability, and built-in security, governance and compliance, functional scalability turbocharges the development and adoption of large-scale, sophisticated blockchain applications, for realizing significant societal impact by these apps heavily depends on overcoming the scalability limitations inherent in smart contracts.

A package-oriented software framework refers to a design that emphasizes the use of packages or modules as functional organizers of code. It entails key characteristics such as modularity, dependency management, code reuse, collaboration, and versioning, etc.

A package-oriented Web3 framework extends the same ideas to smart contract engine. It overcomes specific challenges of smart contract development and takes advantage of specific benefits of blockchain networks at the same time.

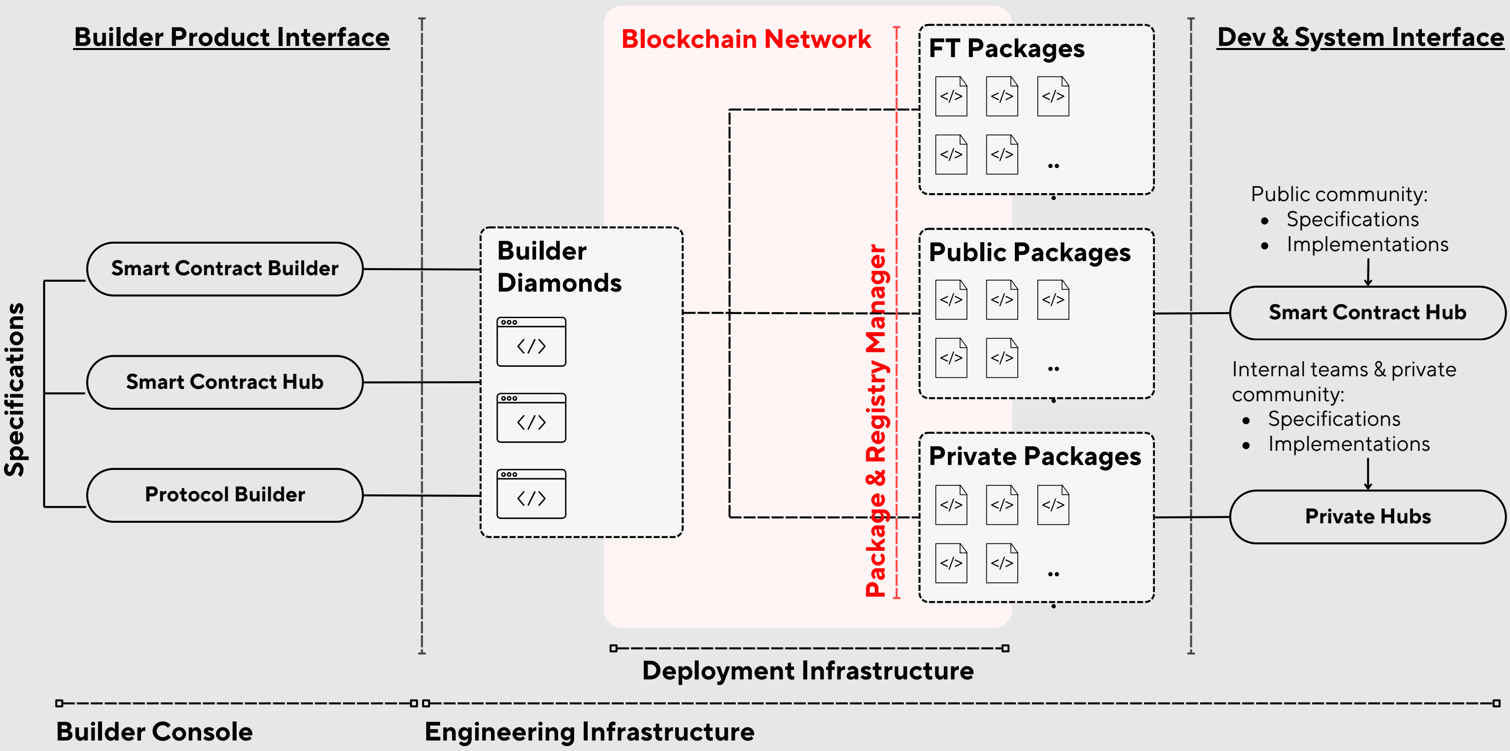
The package-oriented Web3 framework implements modularity. It breaks down the Web3 application into smaller, more manageable packages, which in turn consist of (often composable and reusable) smart contracts. These packages and smart contracts are independently maintainable and upgradable. An additional benefit of modularity is the ease with which codes can be organized, which has broad community appeal.

In order for a package-oriented Web3 framework to gain mass adoption, it has to be designed with certain rules and standards that optimally enforces composability; it should also provide the tooling for project builders to leverage expertise in the community so as to achieve functional scalability quickly and at low cost.

Packages in such a framework can encompass a wide range of components, including on-chain elements, off-chain oracles, and multi-chain components, etc. Packages can be made either private or public. Private packages can be developed by internal teams of the project builder or by an external contractor. Public packages can also be made by either internal or external developers; the difference lies in the choice to make them publicly available to future builders. A valid option is to make public packages open-source, too. This is usually advantageous when there is strategic value in setting a popular standard.

A smart contract hub can be the central piece in building the ecosystem for the Web3 framework. Such a hub functions as a global marketplace of packages. It will support not only packages offered by the developers of the framework but also public as well as private packages.

## Overview of FeverTokens' Smart Contract Engine



The central design consideration for a package-oriented Web3 framework is the smart engineering based on the Diamond Standard (EIP-2535), which is an innovative smart contract architecture. The idea is to divide features across multiple smart contracts called facets. **To achieve functional scalability, heavy engineering on the Diamond Standard is required** for the smart contract engine, hub, and protocol builder with community support.

## Open-Sourcing as Community Strategy

# Path to Monetization

Web3, much like operating systems, cloud computing, and language models in AI, refers to features and functionalities rather than a fixed set of rules and technologies. As such, its standards can only be set by the market.

To succeed as the fabric of web3, a package-oriented Web3 framework built for functional scalability should optimally adopt an open-source strategy that captures market share, increases technological stickiness, and positions the framework as the leader in advanced architectures with premium tooling and services.

Open-sourcing lowers the barrier to entry for builders, underpins community development, improves resilience for critical infrastructure, enhances trust and transparency, and helps create a sustainable ecosystem for package-oriented frameworks.

Combined with the natural advantages of open-sourcing, functional scalability makes building on such Web3 framework a wise option for small and large organizations alike: Small players can push full-featured Web3 products that grow with their teams, while large players ensure that the services that they deploy can be extended with foreseeable as well as unforeseeable features in future.

## How does open-sourcing improve technological stickiness?

First, **customization** helps stickiness. Open-sourcing allows project builders to customize and adapt packages to realize unique features and functions.

Second, **improved control** helps stickiness. The project builders can modify and extend their apps as necessary, and the technology will be viable even when the developer of the framework discontinues support.

Third, **technological inertia** helps stickiness. Over time, organizations will accumulate substantial investment in the framework, both in terms of apps deployed and in terms of knowledge and workflows.

Fourth, **network effect** helps stickiness. Open-source Web3 framework that is widely adoption will have strong network effects. The more users and contributors a project attracts, the more valuable it becomes, and this value can make it stickier for existing users.

Hence, open-sourcing as a community strategy addresses a key obstacle in mass adoption of Web3: Originally, initial cost is high, yet economic efficiency is achieved only at scale. With this obstacle removed, we can expect accelerated adoption that makes the open-source, functional scalable, and package-oriented framework the de facto standard.

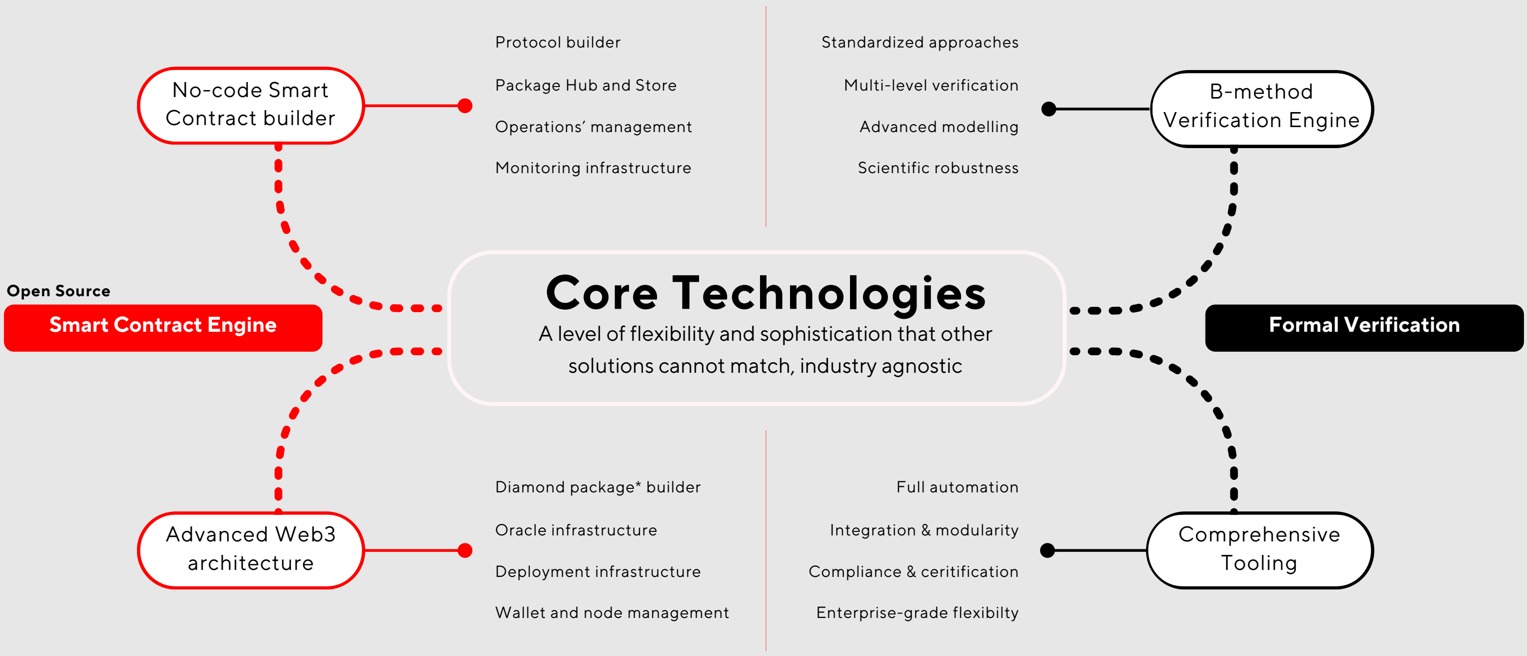
The engineering behind such a sophisticated framework does not come without cost, and indeed many community developers also have their own financial interests to consider. Premium tooling and services integrated to the Web3 framework facilitate monetization while further improving the attractiveness of the framework to project builders.

Because these premium tooling is co-developed with, and seamlessly integrated into, the open-source framework, it offers a natural advantage to FerverTokens as the framework developer in terms of mastering advanced architectures, diamond-powered smart contracts, and related services, including the capacity to oversee community contributions and product offerings. Hence, the framework, though open source, lays the foundation for us to seek appropriate financial reward for the engineering efforts.

Formal verification is another important part of premium tooling and services. It underpins the smooth functioning of the smart contract hubs and facilitates third-party contribution to the ecosystem.

## Integration with Formal Verification

Formal verification with the B-method ensures automated and mathematically-proven consistency and security verification of advanced smart contracts and web3 architectures. Such a verification goes beyond on-chain package-level smart contracts, extending the verification to application wide governance and critical off-chain components.



** Packages are powered by engineered Diamond Standard (EIP-2535) facets and other core implementations*