FORCEN

WHITE PAPER

DEFI

The Decentralized Finance Protocolhe
first layer 1 protocol specifically built to serve Defi

20 August
2020 v1.0

Catalogues

01

s u m m a r y

State

Several papers have been issued by the FORCEN Foundation Limited (Foundation). This is a concept paper to describe the expected future financial characteristics of a cryptographic economic ledger.

Any system initiated by FORCEN Foundation or FORCEN Foundation will be conditional upon the receipt of favorable legal and regulatory advice, approvals and/or licenses from all relevant authorities and jurisdictions.

Neither the FORCEN Foundation nor any related person will be liable for any direct, indirect or consequential loss or damage suffered by any person as a result of reliance on any statement, information or advice contained herein or any comment made in connection with this document.

Blockchain technology is the underlying technology for building the Bitcoin blockchain network and encrypted transmission of transaction information. It is based on the principles of cryptography rather than on credit, allowing any agreed parties to pay directly, thus eliminating the need for third-party intermediaries.

Purpose

We believe that before DeFi can truly go mainstream, Forcen, the new generation of decentralized financial protocols, must establish an open, collaborative, distributed and secure trusted financial ecosystem. The establishment of a trusted financial ecosystem is key to Forcen's technical implementation, application implementation and governance implementation.

Therefore, Forcen values the gathering of wisdom: forcen will be led by a leading blockchain technology team, and will fully integrate the technical expertise of the local and international community and various partners.

Therefore, Forcen values application support: as a DEFI-supported infrastructure, Forcen can provide the technical infrastructure for distributed services for a large number of application scenarios, allowing service delivery teams across industries to conveniently provide distributed services in the financial sector without having to master blockchain or other underlying technologies.

Therefore, Forcen attaches importance to the expansion of the ecology: Forcen is committed to building a cross-chain, cross-system, cross-industry, cross-application and cross-terminal distributed trust financial ecology. Financial Ecology.

02

Protocol Realization

Introduction to the FORCEN protocol

Definition

The aim of the Forcen protocol is to create a bridge to connect the decentralized financial system. As the underlying blockchain infrastructure platform, Forcen will play the most fundamental role in the Defi direction process, i.e. versatility.

In the process of universal operation, Forcen takes into account the different blockchain practices and applications in different scenarios, mechanisms and interactions to meet the different needs of Defi's applications, i.e., customization.

From a technical point of view

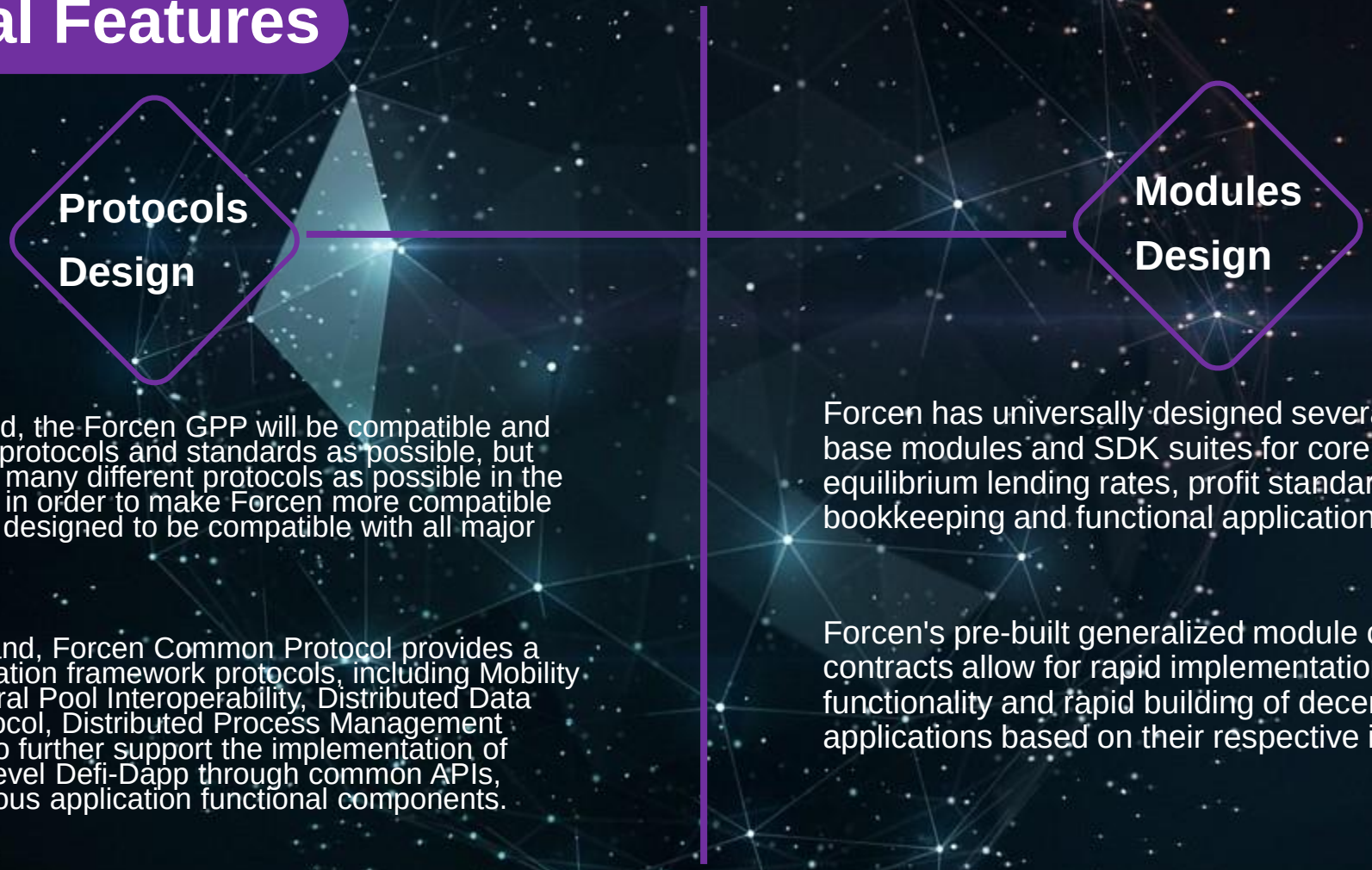
Forcen is not a single technology, but the result of the integration of multiple technologies. These technologies come together in a new structure to form a new way of recording, storing and representing data. This approach has both commonalities and characteristics.

From solutions

Forcen addresses the security, interoperability, and motivation of Defi: i.e., reducing smart contract invocation failures, exploits, and the underlying technical building blocks that facilitate Defi-Dapp.

Protocol Realization

Universal Features



On the one hand, the Forcen GPP will be compatible and adopt as many protocols and standards as possible, but also support as many different protocols as possible in the same scenario, in order to make Forcen more compatible and extensible, designed to be compatible with all major protocols.

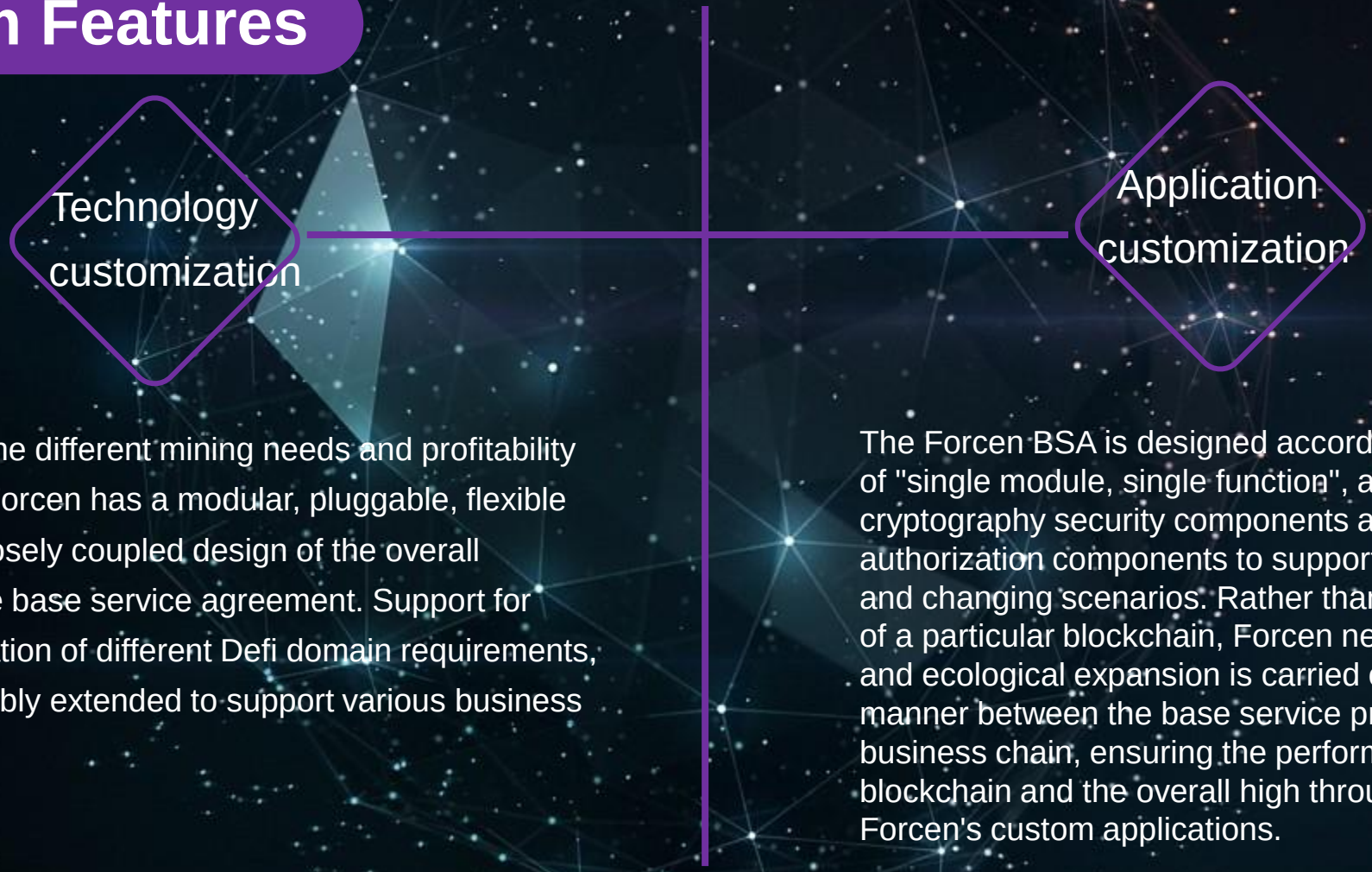
On the other hand, Forcen Common Protocol provides a series of application framework protocols, including Mobility Common, Mineral Pool Interoperability, Distributed Data Exchange Protocol, Distributed Process Management Protocol, etc., to further support the implementation of various upper-level Defi-Dapp through common APIs, SDKs, and various application functional components.

Forcen has universally designed several generalized base modules and SDK suites for core market indices, equilibrium lending rates, profit standards, core bookkeeping and functional applications.

Forcen's pre-built generalized module calls via smart contracts allow for rapid implementation of generic Defi functionality and rapid building of decentralized applications based on their respective industry sectors.

protocol realization

Custom Features



In order to meet the different mining needs and profitability of different Defi, Forcen has a modular, pluggable, flexible and extensible loosely coupled design of the overall architecture of the base service agreement. Support for Forcen customization of different Defi domain requirements, which can be flexibly extended to support various business applications.

The Forcen BSA is designed according to the principle of "single module, single function", and works with cryptography security components and user authorization components to support Forcen's flexible and changing scenarios. Rather than increasing the load of a particular blockchain, Forcen network functionality and ecological expansion is carried out in a collaborative manner between the base service protocol and the business chain, ensuring the performance of each blockchain and the overall high throughput capacity of Forcen's custom applications.

Universal Module Orientation

Smart contract

Commonly designed business rules for each business associate in each system of distributed identity and multidimensional liquidity provider authentication, distributed trust incentive delivery, distributed data exchange, distributed profit sharing, etc.

Defi-Dapp Foundation Components

Designed to work with Forcen's rich set of application layer protocols and components, the Forcen Foundation Services Protocol helps application developers quickly build decentralized financial applications across industries without the complexity of DEX distributed ledger interactions and, based on Forcen's network architecture, allows for better cascading and segmentation of performance scaling, governance models and specific business requirements, and better resiliency support.

Customized Module Orientation

Smart Contracts

As an example, Forcen's customized smart contract system can be designed to support the process-oriented expansion and innovation of Defi's business, establishing specific technical trust mechanisms in mining collaboration, liquidity control and exchange.
Based on Forcen customization, partners can meet the specific technical application and governance needs of each exchange, trader and system, while at the same time being able to integrate with Forcen General and work with common protocols, modules and rules.

merge

Forcen General and Forcen Custom Convergence, which together complete and improve the Forcen network architecture to provide better concatenation, partitioning, and better resiliency support for performance scaling, governance models, and specific business needs. To develop and create a better Defi-Dapp.

03

Core Issues

FORCEN Underlying Implementation

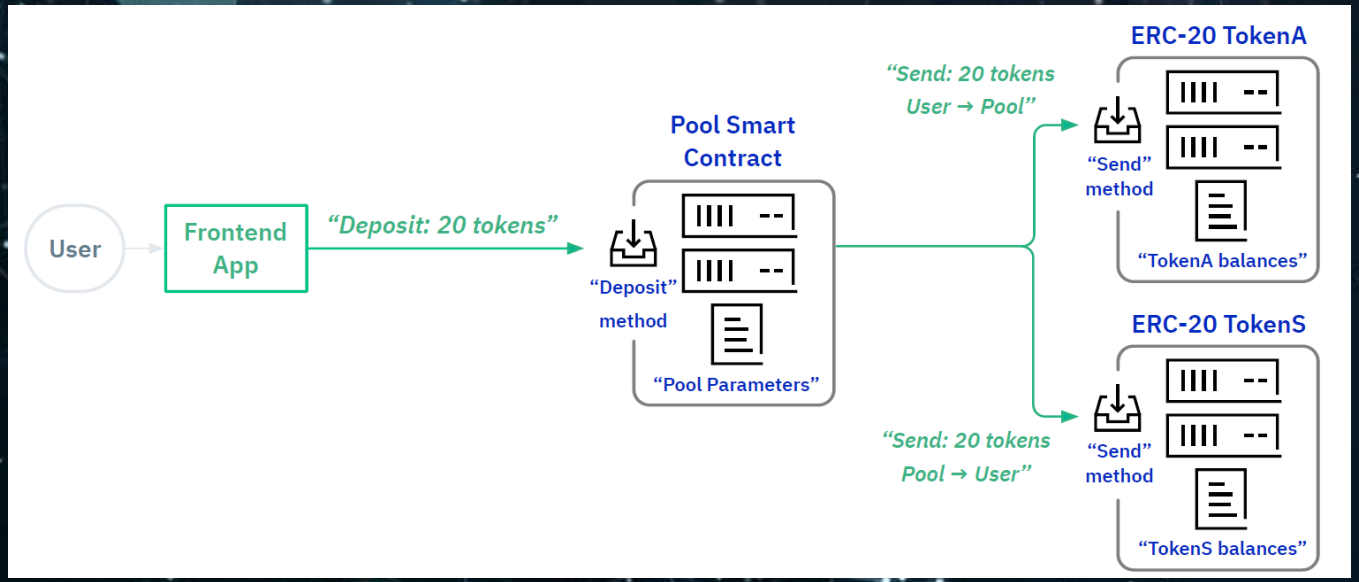
Engine

The Forcen-NLP engine development environment is specifically designed to create logic that defines predictable and correct results on the ledger in response to requests. This form of DLT programmability is based on (FSMs), a class of solutions commonly found in mission-critical embedded systems, where predictable correctness is the first priority. In order to clearly differentiate it from traditional Ethernet-based smart contracts, we have given NLP engine smart contracts a component that better reflects their functionality. This is critical for DeFi, where potentially expensive unintended outcomes are untrusted accounts. The situation is further complicated in DeFi applications where a single transaction involves multiple combined smart contracts. In this case, one contract can invoke methods on other contracts, each of which updates its own internal variables to produce a combined result.

FORCEN underlying logic

Even a simple liquidity "pool" of smart contracts can become complex, with results spread across multiple contracts. If we want our pool to accept an existing (ERC-20) token a into the pool and generate a corresponding calculated number of a tokens, representing a share of the pool, then we will get the following result.

Engine



FORCEN Assembly

Assembly

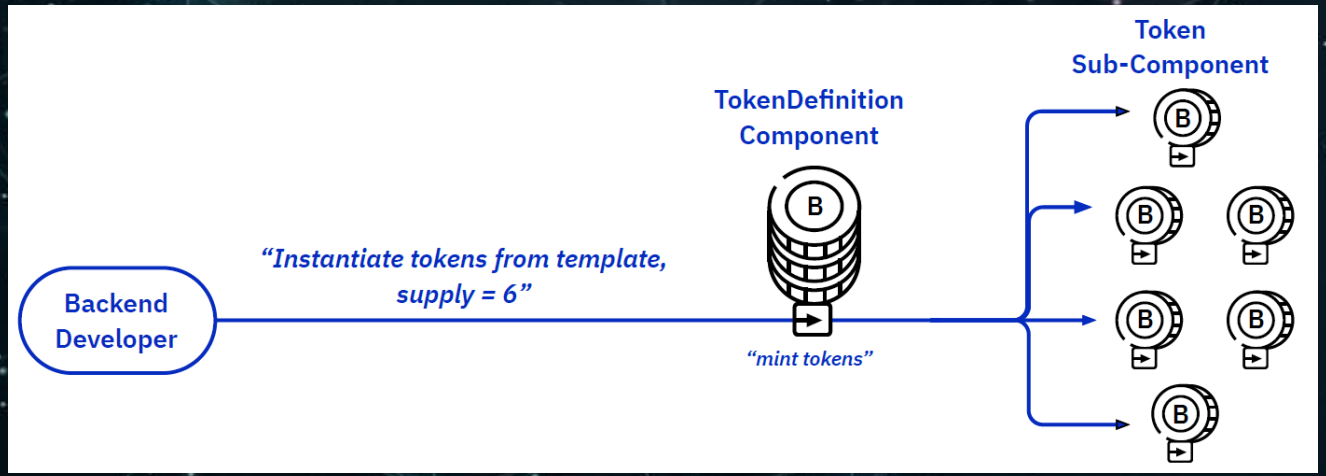
Forcen creates new components using a new proprietary language that we call Scrypto, a functional language that provides a programming style better suited to defining the FEN-based components described here. Functional languages are becoming increasingly popular, especially when building reliable, highly concurrent systems.Scrypto's syntax should be familiar to developers who have worked with functional languages, and provides a set of programming languages that are particularly well suited to creating component/operational logic.

We know that we need to configure and create a token provisioning before we can start sending tokens. And an important feature of using components to do this: the ability to define sub-components. These are components that operate independently (they have their own operations and separate state), but are defined as the top-level "parent" component. User-created tokens are defined internally in this way. Individual token definition components describe the properties of the entire token provisioning (e.g. name, symbol, maximum provisioning, etc.), but also include the definition and behavior of the sub-components.

FORCEN Assembly

Developers use TokenDefinition to customize its parameters. The TokenDefinition component then provides the "mint tokens" operation, which creates indivisible subcomponents of individual tokens. These individual tokens are created via the "mint tokens" operation on TokenDefinition, which creates a copy of the token from the sub-component "template". The tokens can then be used as standalone physicallike objects, accessible through the API, while still being associated with the parent token definition and its configuration. The logic is as follows.

Assembly



Requests for FORCEN matters

Interactions

To understand how users of component operations can obviously prevent erroneous results, let's take the example of a token exchange component. This component accepts TokenA and returns TokenB, which is equivalent to the market price (probably provided by oracle or some market maker logic). Its main purpose might be to：

Exchange Operation: "Any reception of TokenA will create a sending TokenB calculated as TokenA*[the currently defined B/A ratio]"

User request: send the operation "I am TokenA to the "exchange" token exchange component, which must result in a token B that is at least [required limit] for me".

Get Price Action: when requested, I provide the current content of the key-value pair representing the requested pair of prices

Instantiation of FORCEN components

Instances

Instantiation is simple and done through the API, no script code is required. Most of the components in the directory will contain configuration parameters that allow custom instantiation of the component. For example, the "Token Definition" component allows anyone to instantiate their own token definition with their own unique name, symbol, maximum supply, etc., and then start casting their own token. Developers can easily write basic in a generic way without using any of the generic components created by Scrypto; developers wishing to create their own special-priced Oracle components can import the original Price Oracle component by referencing its ID in the directory and add additional Scrypto! Code to calculate the average price.
Catalog components and instantiated components have their own unique component IDs associated with the creator's own unique developer ID. Components are versioned, and a version conversion is required for each newly deployed update. Updating a component does not automatically force the use of its other components for updates; previous versions remain immutable on the ledger, and existing components will continue to have access to previous versions. Developers can choose to adopt the new revision by making their own updates to the components that use it.

04

Application Ecology

FORCEN Application Components

The Forcen Trusted Financial Ecosystem is divided into the Forcen Base Layer, the Component Module Layer and the User Layer. There is no doubt that building the Forcen Trusted Finance ecosystem requires the concerted efforts of many ecosystem partners. At the same time, each level of the Forcen architecture is made up of various eco-partners working together.

Ecological User

Forcen Users

User Authentication

Forcen can be configured to integrate various DEX authentication mechanisms and liquidity provider authentication endorsements, resulting in a multi-dimensional authentication service that will enable multi-source, trusted and secure authentication for users of Forcen's various applications.

User Licensing

Forcen's authorization mechanism, designed to implement any transaction involving data relating to a data subject that requires notification of the data interest (one or more parties) to authorize the transaction 。

Data coordination

mechanismForcen will assist decentralized financial demanders, mining data subjects, centralized management database and other data collaboration, so that each subject can query the location of the required data and carry out corresponding data request operations, thus avoiding the existence of various data islands and broadening data access channels. 。

Ecotechnologists

Forcen Technical Participants

Development Team

The Forcen development team will be responsible for the development and maintenance of Forcen's underlying blockchain architecture framework, and will coordinate with various technology partners for the design and development of related technical modules.

Technical Community

The Forcen Technology Community is an open source technology community that will provide development, testing, innovative design, and related application experience, feedback, and application promotion for Forcen.

Partnerships

Partners are a key force in connecting the underlying blockchain to the application layer. Partners participate in and implement the development, integration and maintenance of their relevant blockchain applications based on Forcen's underlying blockchain architecture.

Eco-Applicants

Forcen Application Participants

Mobility Certification Provider

The Forcen development team will be responsible for the development and maintenance of Forcen's underlying blockchain architecture framework, and will coordinate with various technology partners for the design and development of related technical modules.

Technical CommunityForcen

The Technology Community is an open source, open source technology community that will provide development, testing, innovative design, and related application experience, feedback, and application promotion for Forcen.

Partnerships

Partners are a key force in connecting the underlying blockchain to the application layer. Partners participate in and implement the development, integration and maintenance of their relevant blockchain applications based on Forcen's underlying blockchain architecture.

Ecological Existing Team

FORCEN's current team

Software Developer - Rodrigo Acosta, blockchain developer with extensive experience in smart contracts, assets, bitcoin, Android and governance systems.

°

Software developer - Ivan Vyatkin (Ivan Vyatkin) with 6 years of IT experience. Involved in different types of projects: online stores, account aggregation (Krawlly), etc. Since 2014 Ivan has been working on PWA (Progressive Web Application) development.

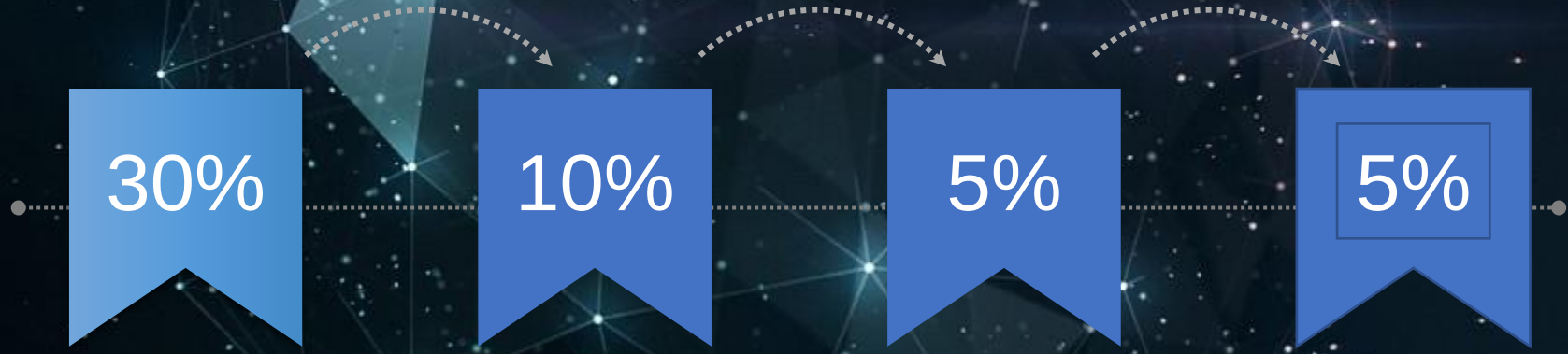
Anton Filatov is actively involved in many different IT projects, from financial systems to distributed systems. He has extensive experience in software development using modern programming languages and advanced technologies.

Nicola Marangoni has strong competencies in analytical databases and distributed systems. Since 1999, he has been involved in multiple data warehousing, business intelligence and analytics projects in many different branches.

Ecological Token Model

FORCEN

The FORCEN token, abbreviated as FEN, is an BEP-20 utility token that will be acquired through a private placement. A total of 100,000,000 tokens will be available with an initial revolving supply of 40 million and the rest will be used to provide liquidity, system development, team operations, marketing, etc. respectively (refer to the official website).



Distributed through pre-sales, whitelisting, and airdrops, all remaining will be destroyed if sales are not completed within the expected time frame.

Used to provide liquidity

Primarily used for ecosystem development, component model design and protocol iteration, development of Defi-Dapp infrastructure and applications.

Team operational use, including but not limited to marketing, collaboration, and technical consultant incentives.

Ecological Route

FORCEN

The FORCEN token, abbreviated as FEN, is a BSP-20 utility token that will be acquired through a private placement offering. The total quantity is 100,000,000 pieces, and 40% of the initial circulation is 40,000,000 pieces.

Private sale: 10,000,000 pieces at $0.05

Public sale: 30,000,000 pieces at $0.06

05

Case Escriptions

NLP Engine Case

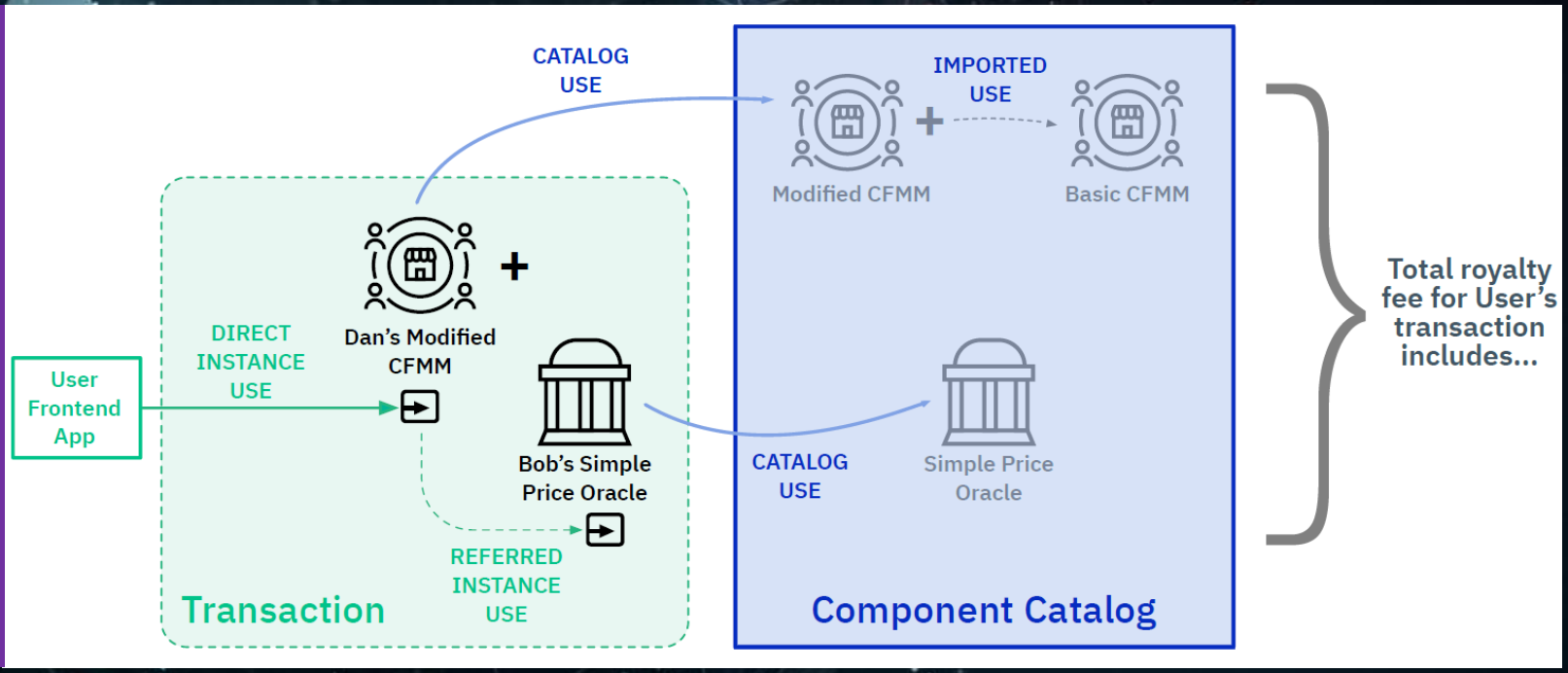
Forcen-NLP request

What NLP wants is a hybrid solution: simple transactions should be parallelized across as many Cerberus shards as possible to maximize throughput. Complex transactions involving a large number of correlated states (e.g., high-capacity DeFi pools) should allow them to be serialized to update the state collected on several (or one) shard as quickly as possible.The NLP engine makes this hybrid solution possible through two elements: action-based requests and dynamic positioning. Case in point:

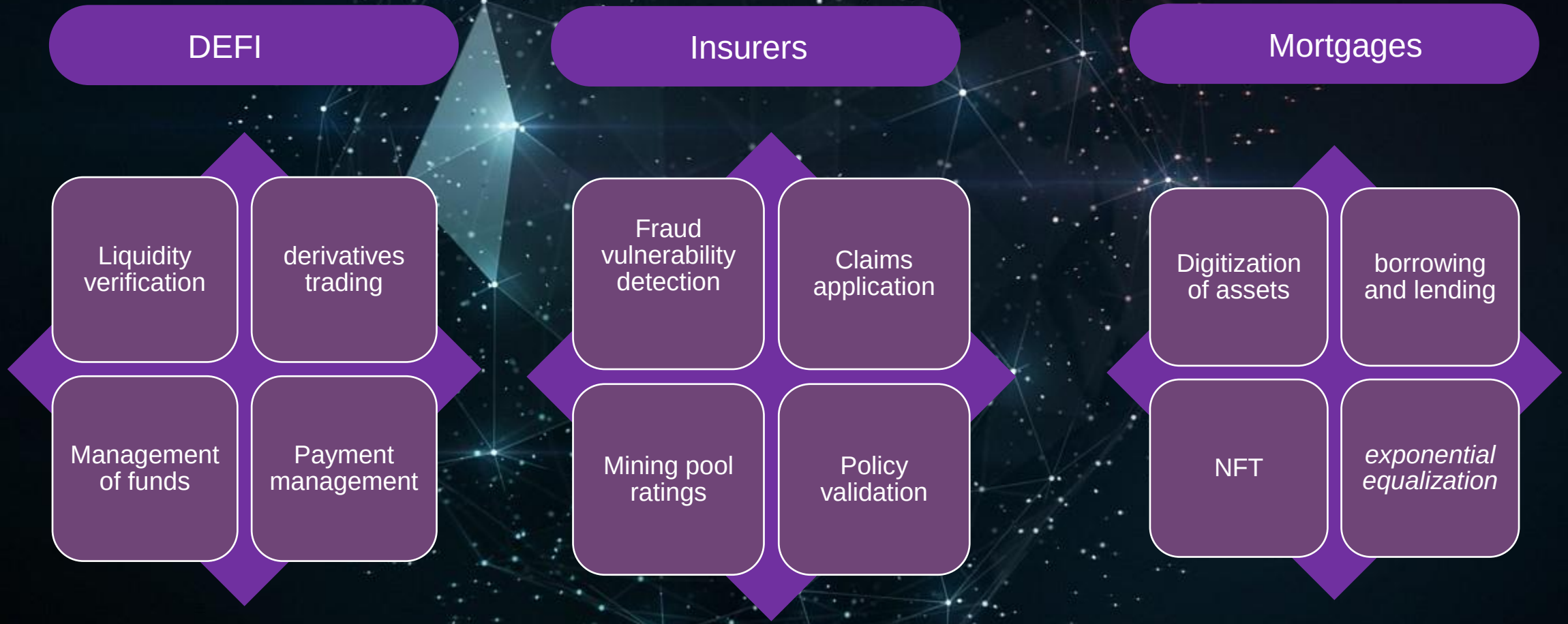
NLP Engine Case

NLP's request for optimism

Depending on the type of use, a user's transaction request for the operation of a modified CFMM (and instantiated component) must include the total usage fees for the component in question. These total fees are easily determined by the base engine, so the user application knows what it must pay and ensures that all fees are paid correctly.



Scenarios


DEFI
Liquidity verification
derivatives trading
Management of funds
Payment management
Insurers
Fraud vulnerability detection
Claims application
Mining pool ratings
Policy validation
Mortgages
Digitization of assets
borrowing and lending
NFT
exponential equalization

Scenario 1

Effective synergy of trusted data

FORCEN allows for efficient collaboration of data sources through the NLP engine when performing user portraits of different liquidity providers. First, there is no change to the existing data management model. By accessing FORCEN, the relevant request data source can still adopt the existing centralized data aggregation model. Secondly, the data collaboration mechanism. When the data demand side queries the location of the required data (i.e., data source), the corresponding data request operation can be carried out. Through data collaboration, a more comprehensive data integration is carried out to facilitate the formation of a holographic credit portrait of the DEFI.

Scenario 2

Chain management of funds

- Through the use of FORCEN, we will achieve low cost and efficient management of funds.
- First of all, the entire process of funding is documented. From the application, approval, and repayment of liquidity funds, etc.
- A multi-party-maintained blockchain system is used for the credible deposit of money processes.
- Secondly, digital money management. With FORCEN, digital funds associated with the blockchain can be defined. The digital funds are combined with the relevant offline processes so that each fund can be viewed, audited and reconciled in real time to ensure the safety of the funds.

Scenario 3

Mechanisms for the authorization of data

The flow of trust and discharge of the whole process of chain records, using the openness and transparency of the blockchain, to protect the trustworthy records and supervision of the flow of rights and interests. In possible future disputes over rights and interests, reliable evidence information can be extracted based on FORCEN to protect the legitimate rights and interests of rights owners.

06

Future Plans

Planning Principles

FORCEN, as the infrastructure and connector of the Trust Financial Ecosystem, will enable the development and collaboration of various applications for various scenarios. The following development principles will be upheld in the development of Forcen's application ecosystem.

Flexibilities

The Forcen architecture is designed to be modular, pluggable, flexible and extensible in a loosely coupled manner to support the needs of different financial business domains and can be flexibly extended to support various business applications.

Innovativeness

Forcen will always combine the development of relevant technologies in the field of decentralized finance, blockchain and distributed ledger, and the development of innovations in application development in different business areas to provide the infrastructure for different application projects to be applied in various business areas.

Groundedness

The Forcen architecture architecture can support the creation of different types of blockchain applications, and the Forcen team will provide technical support in the direction of application landing as the Forcen application ecosystem moves forward.

Cooperation And Exchange

In order to build a collaborative technology ecosystem, the Forcen Technology Ecosystem partners will communicate and collaborate in a variety of ways and will include, but not be limited to, the following.

technical exchange

Based on the technical power of Forcen's "two wings", Forcen will organize various forms of technical communication with the technical community and partners to discuss the development direction and update needs of the Forcen architecture.

Design of underlying technical facilities.

With the development of blockchain technology and the needs of different business fields and regions, the technology ecology needs to keep pace with the times and discuss the various protocols and modules that need to be designed according to the core distributed ledger, smart contract system, and security system.

Application Framework Design.

As Forcen-based applications expand, there will be an ongoing need to develop and update the underlying blockchain application framework, and the Forcen technology ecosystem will jointly explore application components, protocols, SDKs, and APIs that need to be developed or technically iterated in order to meet application developers' sustainable decentralized application implementations.

Technological Innovation

In order to build a collaborative technology ecosystem, the Forcen Technology Ecosystem partners will communicate and collaborate in a variety of ways and will include, but not be limited to, the following.

Technical collaboration

Forcen's technology partners have different technical capabilities and experience in technology development. Forcen welcomes partners to contribute to the design, development, and testing of specific protocols and modules based on their needs.

Development collaboration.

Technical eco-partners can join Forcen's development program to contribute code, review code, provide testing, etc., based on Forcen's technical development needs.

Motivational development.

Based on the specific application-specific needs of Forcen's application partners, especially where custom development is required, Forcen will design incentivized development requirements to solicit technical contributions from the technical ecosystem.

Conclusion

This white paper designs the path and way to build a trust financial ecology based on technology ecology, governance ecology and application ecology.

Establishing a trust financial ecosystem is a huge and arduous task. Therefore, Forcen hopes that all kinds of technology partners, application partners and all kinds of contributors and volunteers will join and contribute their own strength.

At the same time, Forcen also hopes that through the joint efforts of ecological partners, we can bring trust to everyone, so that decentralization is not so far away.