ORBS

Technology Overview

Orbs is a public blockchain infrastructure designed for enterprises. Unlike private and permissioned blockchain solutions typically used for enterprise, the Orbs infrastructure is open and permissionless. The Orbs protocol is decentralized and executed by a public network of permissionless validators using Proof-of-Stake consensus.

The Orbs protocol relies on the ORBS token used for the settlement of fees related to app execution and provides the system of incentives used to elect validators in a secure and decentralized manner.

SERVERLESS CLOUD FOR APP DEVELOPERS

The Orbs platform is a decentralized serverless cloud allowing app developers to build backend services (apps) that are then deployed to be executed by Orbs validators. The main offerings of the platform include compute under consensus (execution of “smart contracts”) and storage under consensus.

These apps are unique in their ability to facilitate trust by providing blockchain-backed guarantees to their users. Whereas many popular public blockchain solutions like Ethereum are designed primarily for purely decentralized apps (dapps), Orbs expands the offerings of public blockchains by supporting permissioned apps developed by existing for-profit businesses.

COMPLETE BLOCKCHAIN STACK

The Orbs project includes a full blockchain stack designed and implemented from scratch by the core team, and is not a fork of any existing solution. The project is active on open source with over 60 repositories publicly available on Github.

The codebase provides an end-to-end developer experience with everything needed to create and execute blockchain apps including: microservice implementation of the node core; consensus algorithm (Helix); smart contract SDKs for building backends in multiple languages; local development server (Gamma); block explorer (Prism); client SDKs for building web and mobile frontends in multiple languages; and more.

FURTHER READING

- Overview of Orbs project white papers
- The Orbs Blog on Medium
- Orbs open source repositories on Github
- Orbs protocol specification on Github
- ’Blockchain as the Next Evolutionary Step of the Open Source Movement’ post

Hybrid Two Tier Consensus

Today's blockchain infrastructure landscape is polarized where solutions are moving between two extremes: public blockchains like Ethereum - permissionless infrastructure designed for decentralized apps; and private blockchains like Hyperledger - permissioned infrastructure designed for permissioned apps.

Orbs attempts to bridge this dichotomy (see diagram) with a hybrid architecture - **permissionless infrastructure which supports permissioned apps**.

BLOCKCHAIN-BACKED GUARANTEES

Under Orbs, the value of public blockchain is defined as the ability to facilitate trust by creating apps providing blockchain-backed guarantees to their users and partners. These include **auditability** - the ability for users to know the app protocol and audit its data; **forkability** - the ability for users to leave the app with their data; **governance** - the ability to enforce how the app protocol changes.

BLOCK PRODUCERS VS. VALIDATORS

The first step of blockchain consensus is selecting a node to choose transactions and create a block, the second is for other nodes to validate the block and approve it. These steps are typically performed by the same nodes. The Orbs protocol separates these roles to two different node types - **block producers** that create blocks and **validators** that approve them.

VALIDATORS CONTROLLED BY THE NETWORK

Validators are selected by the network from the permissionless pool according to the Orbs Proof-of-Stake incentives model. They actually provide the guarantees to the app's users and partners by acting as an impartial third-party source of trust that is provably unbiased.

BLOCK PRODUCERS CONTROLLED BY THE APP

Block producers are selected by the app. They can be permissioned, and provide the app developer with full control over security and governance. This separation of power limits attack vectors since collusion by Orbs network validators cannot affect created blocks. Block producers can also evolve by the app over time to become permissionless, based on the app's own incentives model (eg. stake with the app's secondary token) or the network's.

FURTHER READING

- 'Defining the Public Blockchain' post
- 'The Blockchain Dichotomy and an Architecture to Overcome It' post

Virtual Chains

Every app running on Orbs typically runs on its own virtual chain. Virtualization provides apps with an isolated environment while utilizing a shared physical infrastructure of nodes. Every virtual chain maintains its own separate chain of blocks, state, and runs its own concurrent instance of consensus.

Every Orbs validator runs all virtual chains in parallel, making the permissionless pool of validators completely shared and staked across the full network. This provides apps with the security and decentralization of the entire network while maintaining a strong degree of independence.

ISOLATION AND DEDICATED RESOURCES

Every virtual chain is allocated separate computing, storage and consensus resources providing a guaranteed SLA to its users. Congestion in one virtual chain does not propagate to other virtual chains, making performance predictable and reliable. Resource isolation also prevents fee surges resulting from competition and enables a predictable fee model absent from most shared-resources blockchains.

INDEPENDENT APP GOVERNANCE

Isolation between virtual chains allows each app to make its own protocol governance decisions, such as fixing a vulnerability in a deployed smart contract or customizing a protocol parameter like its consensus block rate. This provides apps with the stability lacking in shared infrastructure solutions and reduces protocol forks as interests are less likely to conflict. Apps are also no longer at risk of having their block history reverted due to a network-wide decision such as The DAO fork or prevented from deploying a fix like with the Parity bug.

INFINITE HORIZONTAL SCALABILITY

The parallel and concurrent operation of different virtual chains results in inherent sharding, as smart contracts are likely to interact mostly with smart contracts deployed on the same virtual chain. When a new virtual chain is created, Orbs validators dynamically allocate more resources for its execution (eg. instantiate a new AWS machine in the node cluster) allowing for a virtually limitless number of virtual chains to run on the network in parallel.

FURTHER READING

- 'Blockchain Virtualization' working paper
- The Orbs Position Paper (pages 53-60)

Proof-of-Stake over Proof-of-Work

The Orbs Proof-of-Stake ecosystem is the backbone of the Orbs platform and serves as the foundation for the security and decentralized operation of the network, providing a provably unbiased source of trust for apps running on top.

Orbs relies on Ethereum mainnet as an external decentralized source of trust for executing the mechanics of the staking mechanism (placing the votes) and for fee settlements for virtual chain subscriptions. The ORBS token is thus implemented as an ERC20 token over Ethereum.

PROOF-OF-STAKE INCENTIVES MODEL

The Orbs incentives model recognizes three interest groups and optimizes for separation of power between them. **Validators** are professional node operators, rewarded for executing virtual chains using fees and lock stake to participate (regular PoS). **Guardians** are the representatives of stake (DPoS), they actively monitor and audit Validators and participate in frequent elections to select them.

Relying on token holders to monitor the network introduces a challenge of low participation. The third group, **Delegators**, are the long tail of silent stake and are rewarded for delegating their stake to a Guardian for frequently voting on their behalf.

ELECTION OBJECTIVITY WITH PROOF-OF-WORK

The *objectivity* property of Proof-of-Work means that an external observer can objectively verify the correctness of state represented by nodes (by verifying the work and using the longest chain rule). Classic Proof-of-Stake implementations lack this property. Observers attempting to verify state will have to verify that the signing validators were elected properly. Since elections and votes reside on the chain itself, signed by the very same validators, they would encounter a circular dependency.

Orbs sets a higher standard of trust in Proof-of-Stake by relying on an external Proof-of-Work network, the Ethereum mainnet, for processing votes for validators. This adds the objectivity property of Proof-of-Work to Orbs stake-based validator elections and prevents Orbs validators from processing their own elections.

Orbs is not a layer 2 solution over Ethereum as it has its own independent incentives model.

FURTHER READING

- Orbs Proof-of-Stake Ecosystem Model specification
- 'Why Proof-of-Stake Systems Can Benefit From External Oversight' post
- 'The Orbs PoS Universe Archetypes - High Level Introduction' post
- 'PoS Under the Orbs Architecture' post

Helix Consensus Algorithm (RPoS)

Helix is a Byzantine fault-tolerant consensus protocol based on PBFT that provides the security and decentralization of a large set of validators (eg. 1000 nodes) with the speed and efficiency of a smaller set (eg. 22 nodes).

A key element in Helix is a source of common and verifiable randomness which is used in helping obtain both scalability and fairness (Randomized Proof-of-Stake).

RANDOMIZED COMMITTEES FOR SPEED

A fixed-sized committee of nodes (eg. 22 of 1000) is randomly selected per block for its consensus. The committee selection is based on a verifiable random function generated from a BLS unique threshold signature. Only after a block is committed, the threshold signature can be calculated providing the random seed for the next block committee, guaranteeing that selection can't be manipulated or predicted in advance.

RESISTANCE TO CENSORSHIP AND FRONTRUNNING

Transactions can be encrypted by end-users before transmission, ensuring that the ordering of transactions by nodes is over opaque data and thus fair. The transactions can only be decrypted for execution after consensus over order by relying on threshold encryption. The protocol also enforces a fair mixture of transactions propagated by different gateways. These practices prevent manipulations by ordering nodes such as censorship and frontrunning.

NODE REPUTATION SYSTEM

In order to rapidly identify faulty nodes, balance resources and incentivize nodes to behave according to the protocol, the algorithm can maintain a decentralized reputation system where every node is scored by its peers. Reputation affects a node's chance of being included in committees and assists in economic incentivization such as payment of fees to operators.

FURTHER READING

- Helix Consensus Algorithm peer reviewed research paper
- 'Eliminating the Security vs. Scalability Dilemma' working paper
- 'Enforcing Fairness in Transaction Ordering' peer reviewed research paper
- 'Orbs Unified Theory of Randomness: Explaining Helix Consensus' post
- The Orbs Position Paper (pages 50-52)

Polyglot Smart Contracts

As blockchain development becomes mainstream, platforms must offer a lower barrier to entry for blockchain developers to remain competitive.

SMART CONTRACTS IN ANY LANGUAGE

Orbs pioneers the paradigm of smart contracts as a library in any language, like Go and JavaScript, instead of dedicated languages like Solidity. This flexibility allows existing teams to transition to blockchain development and existing tools and libraries to be reused. Smart contracts are deployed on-chain as source code, making them easier to read and thus safer, and compiled locally on-demand by validators. Implementation challenges include efficient sandboxing and dealing with non deterministic outputs that cannot undergo consensus.

NON-DETERMINISTIC EXECUTION

Since execution engines of standard languages are not deterministic in nature (eg. heap addresses), dealing with non-determinism is a core requirement of the platform. Starting from fail-safe mechanisms in the consensus algorithm itself for failing transactions where consensus is impossible, to APIs defining acceptable thresholds for consensus when the results for each validator differ (while storing each validator's result for future audit).

WEB ORACLES AND OFF-CHAIN INTEROPERABILITY

Providing non-determinism as a feature opens up new use cases for smart contracts such as accessing web oracles and off-chain data that isn't straightforward to access under consensus. The benefits are clear as many real-world applications require to interoperate with existing systems or databases that are not on-chain. Native support for direct query of external data by validator nodes obsoletes reliance on dedicated oracle nodes that often become the weak links in the system.

CROSS-CHAIN INTEROPERABILITY

Orbs smart contracts provide seamless interoperability with Ethereum mainnet and allow reading Ethereum state directly using the above mechanism.

FURTHER READING

- 'Architecture Considerations to Compete with Cloud' working paper (page 6)
- The Orbs Position Paper (page 80)
- Orbs Smart Contract SDK on Github

Competitive Cost Model

In order for decentralized architectures to penetrate mass-market applications, platforms need to be competitive on cost with their centralized alternatives. Due to the unique design of the economic model determining usage prices Orbs, users should expect execution costs to be closely correlated with the costs of similar resources on regular cloud platforms like AWS, when taking into account a fixed factor of data replication.

PREDICTABLE MONTHLY SUBSCRIPTIONS

A severe business adoption problem with popular public blockchains is not only that fees are high, but that they are impossible to plan and budget for in advance due to the shared bidding market over limited resources. Since Orbs validators allocate separate compute resources per virtual chain, fees are paid as a monthly subscription correlated to the reserved compute and storage requirements - making fees predictable.

PROGRAMMABLE FEE MODELS

Subscription fees for Orbs virtual chains are normally paid by the app developer that created the virtual chain. This familiar model allows apps to subsidize infrastructure costs of their end-users and to budget monthly for bandwidth instead of the per-transaction gas model prevalent in public blockchains like Ethereum where end-users pay fees directly.

To accommodate for more complex scenarios, smart contracts on Orbs can control the conditions for a transaction to execute thus allowing app developers to program custom fee models and even offset costs directly on users (eg. per-transaction fees using an app token).

MARKET DESIGN FOR SUSTAINABLE LOW PRICING

Public blockchain solutions are often based on a diseconomy of scale where increase in demand yields higher operation fees due to execution resource scarcity. For public blockchain to be competitive with regular cloud, it must retain a similar pricing model where prices remain fixed as demand increases or even drop.

The dynamic allocation of resources per virtual chain under Orbs architecture creates a market where validators do not increase prices with demand (perfect elasticity of supply). In addition, lack of inflation in token supply contribute to a healthy sustainable future where apps fund validators by paying fees for resources actually consumed.

FURTHER READING

- 'Architecture Considerations to Compete with Cloud' working paper (pages 2-4)
- The Orbs Position Paper (pages 53-57, 62-63)
- Orbs operation fees and subscription costs

Miscellaneous

AUTONOMOUS SWAP BRIDGE (ASB)

Orbs provides inherent interoperability with base-layer protocols like Ethereum, enabling the seamless transfer of tokens or assets in an autonomous, atomic, decentralized and secure manner. This feature helps scale existing ERC20 tokens by moving parts of the supply to Orbs.

Autonomous swap enables to transfer tokens or assets from other blockchains to the Orbs platform and vice-versa. When an asset is transferred to the Orbs platform, it is anchored in a smart contract and automatically transferred to the Orbs platform proxy token contract. The transfer is performed in a decentralized manner using efficient proofs without dependency on supply and demand, guaranteeing a fixed ratio, enabling to maintain one token over multiple blockchains.

Further reading: Live feature demo on Github

DECENTRALIZED KEY GENERATION (DKG)

Group and threshold cryptography such as threshold signatures are used by the Orbs platform for RPoS committee random seed generation. These algorithms require distributed key generation capable of running in a Byzantine environment.

Orbs utilizes a distributed key generation algorithm implemented on Ethereum as standard contracts providing incentives, communication and dispute resolution layers for the process. Operating over a blockchain provides a reliable means of performing the process frequently without relying on manual generation ceremonies.

Further reading: 'Rational Threshold Cryptosystems' peer reviewed paper

STATELESS VALIDATION AND EXECUTION ACCELERATORS

Block producers are required to maintain the full synced state for every transaction execution. Validators, however, may maintain only a minimal amount of state required to verify proofs of data validity provided by block producers, increasing network efficiency.

This approach can be leveraged further to increase transaction throughput with the introduction of a high-performance accelerator, a centralized executor providing result proofs that can efficiently be sharded and verified concurrently by lower performance validators. By transforming the problem from decentralized to distributed we can scale the system to reach the performance bounds of the centralized alternative.

Further reading: 'Accelerating Decentralized Execution' peer reviewed research paper

General Note

This document describes the overall vision and current technological plan that Orbs Ltd. has conceived for the Orbs network. A production-ready V1 of the Orbs network was released to the public on March 28, 2019. Not every feature described here has been implemented in the currently-available version of the Orbs network and certain features are currently anticipated to be introduced in the future, subject to further development. In the case of any doubt regarding whether a described feature has been implemented, please refer to the specifications contained in the applicable Github repository.

Actual developments in the future may depend on a variety of factors, including contributions made to the Orbs network by other development teams and the acceptance of proposed changes, or lack thereof, by other participants in the Orbs ecosystem, in accordance with the applicable governance procedures that may be in effect from time to time.

Legal Disclaimer

This document is being provided for informational purposes only and may be subject to change. Orbs Ltd. cannot guarantee the accuracy of the statements made or conclusions reached in this document and Orbs Ltd. expressly disclaims all representations and warranties (whether expressed or implied by statute or otherwise) whatsoever, including but not limited to:

- any representations or warranties relating to merchantability, fitness for a particular purpose, suitability, title or non-infringement;
- that the contents of this document are accurate and free from any errors; and
- that such contents do not infringe any third-party rights.

Orbs Ltd. shall have no liability for losses or damages (whether direct, indirect, consequential or any other kind of loss or damage) arising out of the use, reference to or reliance on the contents of this document, even if advised of the possibility of damages arising.

The information contained in this document is intended for informative purposes only and shall not form the basis of, or be relied upon in connection with, any offer or commitment whatsoever in any jurisdiction, including (without limitation) the United States or the State of Israel. This document does not constitute a prospectus or disclosure document and is not an offer to sell, nor a solicitation of any offer to buy any investment or financial instrument in any jurisdiction.