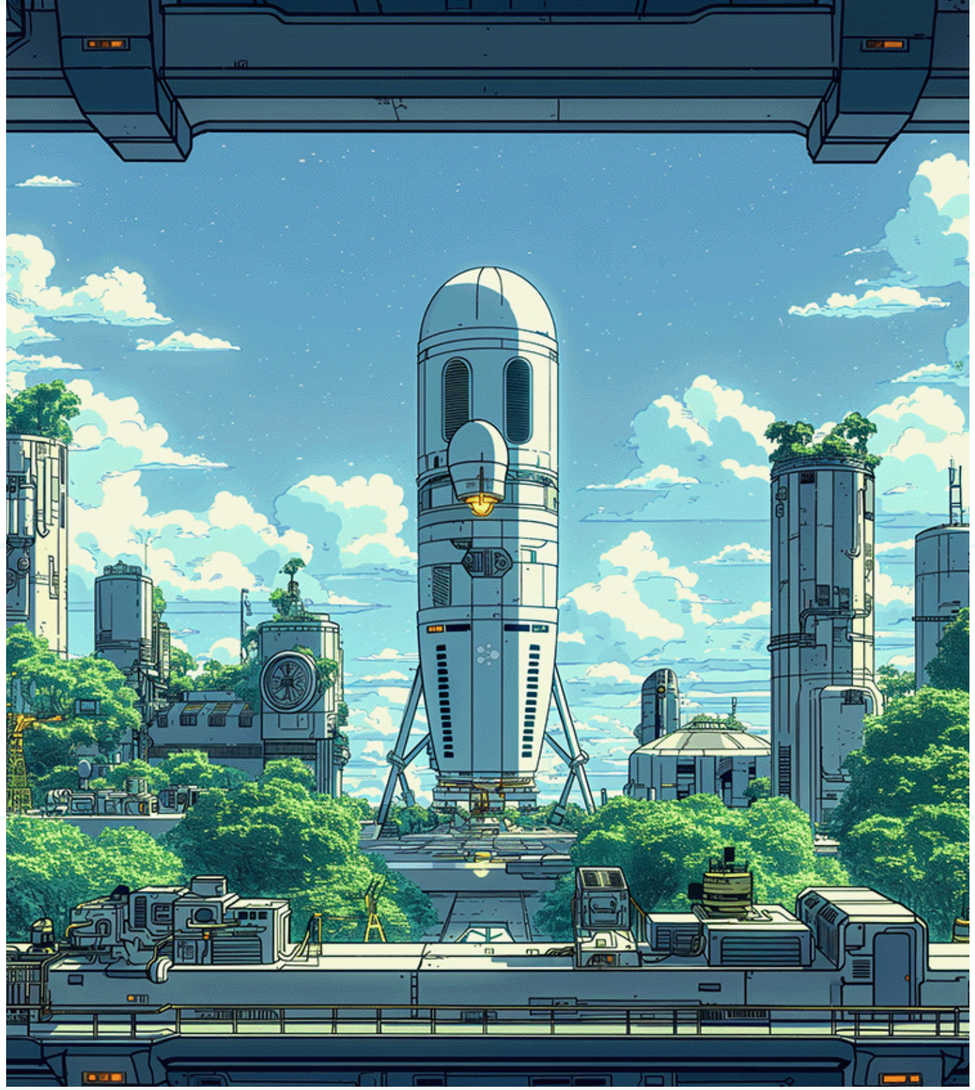
aelf.

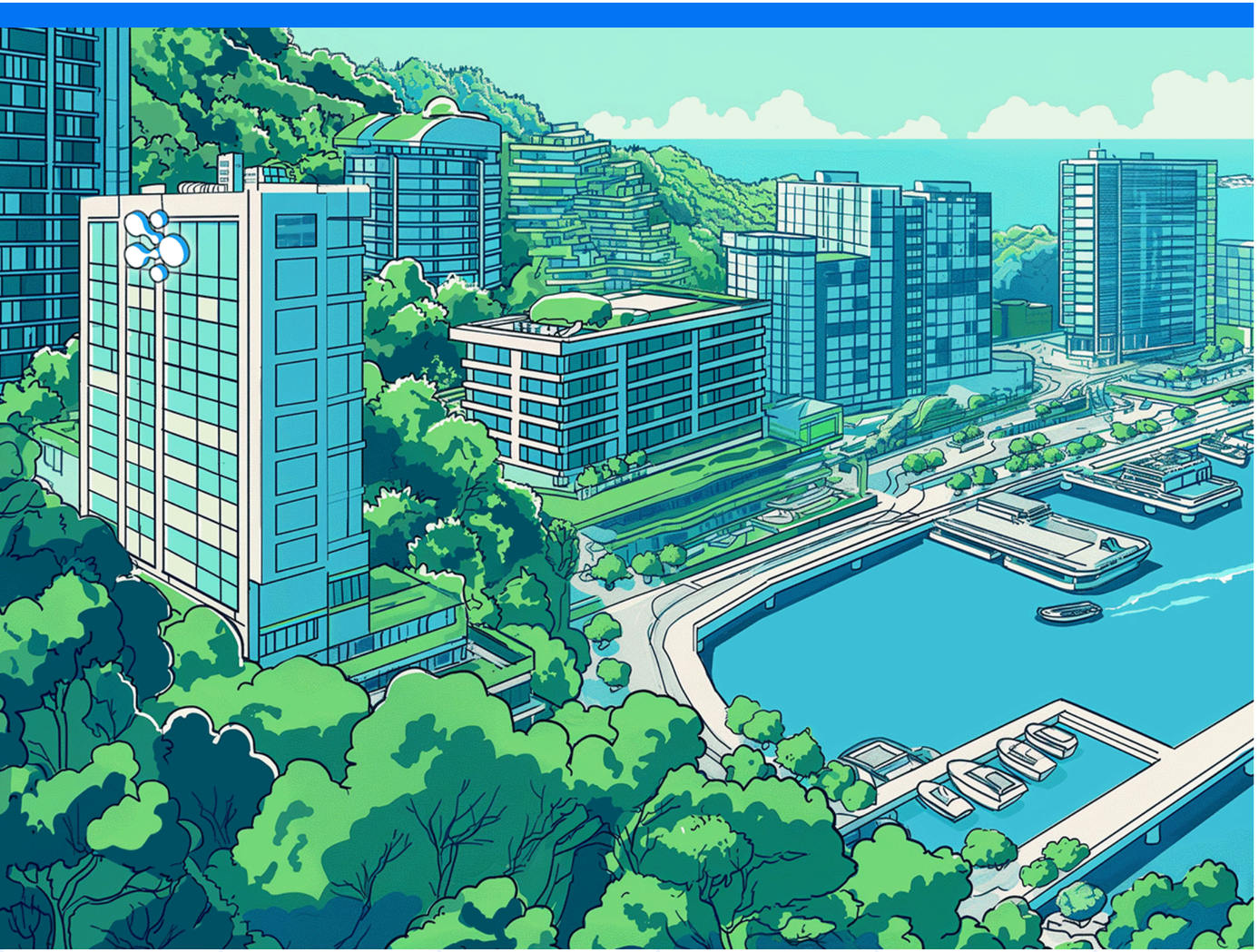
Experience AI + Blockchain of Tomorrow

Whitepaper 2.0



July 2024

Table of Contents

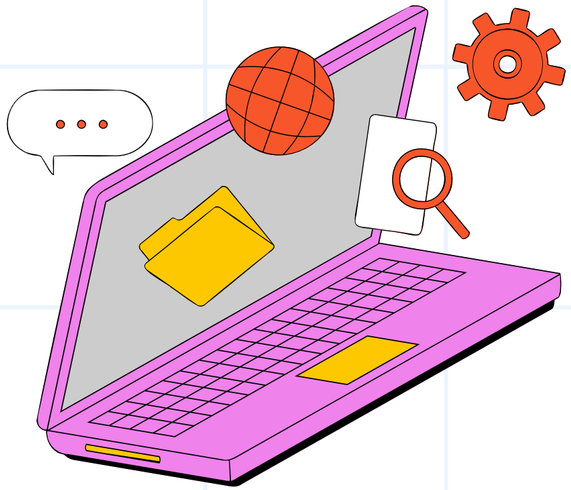


Executive Summary

aelf is a cutting-edge multi-chain blockchain framework launched in December 2017, utilising the powerful C# programming language. Its multi-layered structure, featuring MainChain and SideChains, effectively tackles the challenges of traditional blockchain systems, such as performance bottlenecks and data complexity. This architecture improves scalability and reduces congestion by distributing tasks across multiple chains. With an emphasis on developer customisation and parallel processing, aelf's framework is versatile, supporting a wide range of applications from digital asset management to DeFi.

In aelf's Whitepaper 2.0, the mission is to leverage artificial intelligence (AI) to enhance its operational efficiency. AI-driven machine learning models optimise smart contract execution and gas usage, predict network congestion, and dynamically adjust resources to ensure high performance.

Additionally, AI enables intelligent load balancing and predictive scaling, ensuring efficient traffic distribution and resource utilisation throughout the network. For aelf's AI Oracle, it's aim is to implement a credible Web3 AI framework that enhances on-chain smart contracts with AI features, promoting the creation of AI-centric decentralized applications(AI-dApps).

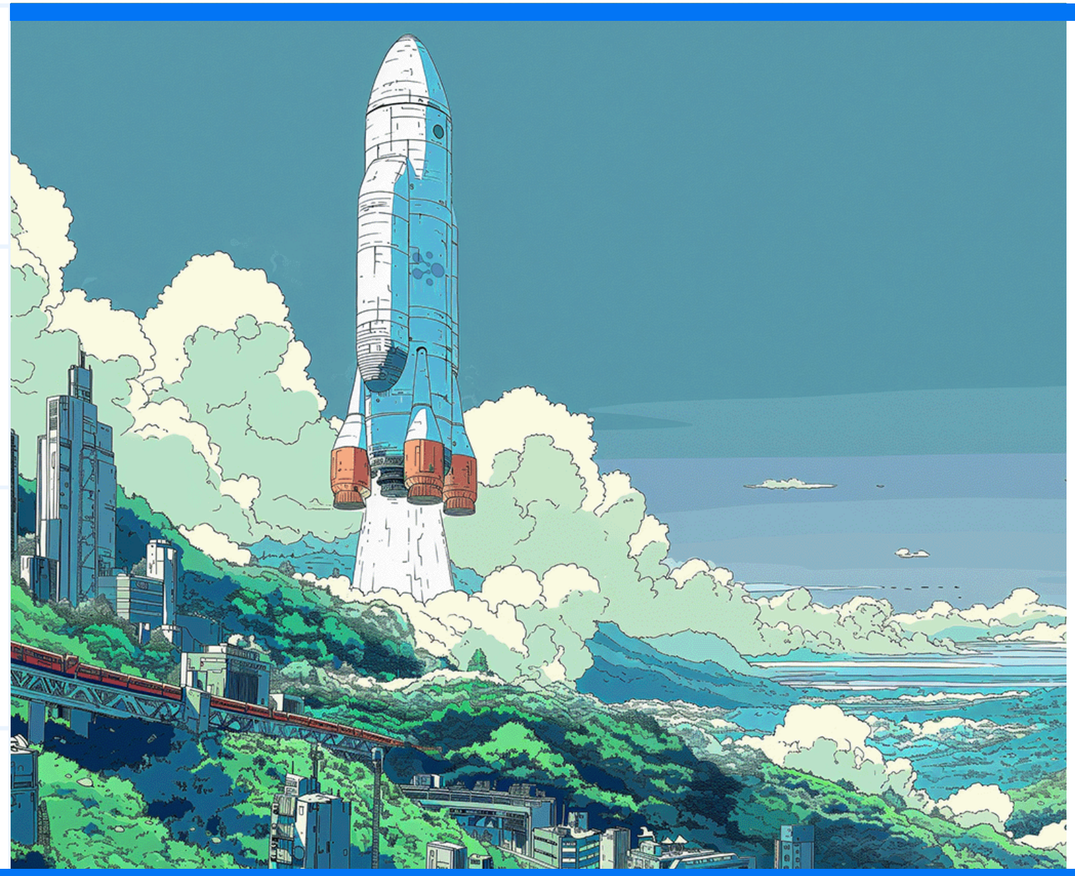


For developers, aelf provides an AI-enhanced environment that simplifies smart contract creation with Natural Language Processing (NLP) and offers real-time support through AI-powered chatbots. These tools make blockchain technology more accessible to non-technical users and streamline the development process for experienced developers. In aelf's AI-based smart contract auditing, smart contract code is input into aelf's pre-trained model to analyse syntax and logical structure, identifying security vulnerabilities and errors, and generating a detailed audit report with remediation suggestions for aelf developers before deploying their smart contracts on the aelf chain. By incorporating AI, aelf not only improves the user experience but also expands its user base, fostering adoption and innovation in the blockchain space.

Beyond AI, aelf will be introducing a new modular Layer 2 chain utilising ZK Rollup technology that would significantly enhance aelf's performance, scalability, and EVM compatibility. This approach will reduce Layer 1 gas storage costs, lowers transaction costs for users, and ultimately improves privacy and the user experience.

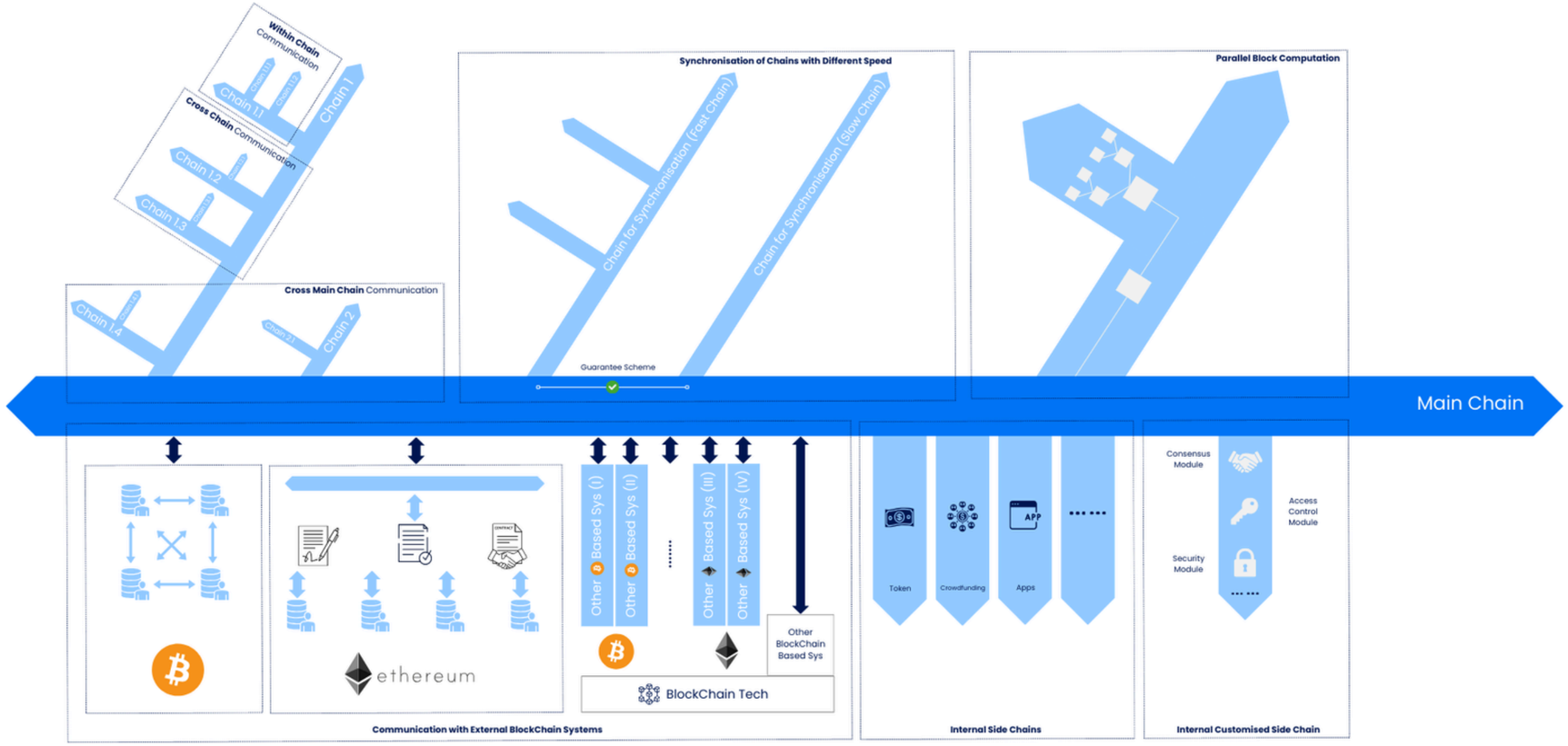
Ultimately, aelf is a versatile multi-chain blockchain framework that enhances efficiency and scalability through a multi-layered architecture, AI integration, and modular Layer 2 ZK Rollup technology, making it highly efficient, secure, and developer-friendly.

Background on aelf



The story of aelf began on 10 December 2017, when aelf's vision and plans were introduced to global investors at a Coindesk conference. aelf successfully completed its fund-raising significantly ahead of schedule, having secured investments from notable institutions such as Binance, Arrington Capital, Draper Dragon, Galaxy Digital etc. In the same month, the ELF token was listed on six major exchanges, including Binance and OKEX. aelf Testnet was successfully launched in 2018, followed by another successful launch of the Mainnet in 2020.

aelf is primarily a multi-chain parallel computing blockchain framework which was created to address several of the limitations of traditional blockchain systems, such as performance bottlenecks and data complexity. aelf's vision is to be a "Linux Ecosystem" equivalent for a blockchain that incorporates state of the art IT design principles, which in particular allows developers to customise Chains to meet their own needs. At its core, aelf contains the following main features:



01

MainChain and multi-layer SideChains to handle various user driven scenarios. One chain is designed for one use case, distributing different tasks on multiple chains, which improves processing efficiency.

02

Communication with external blockchain networks, such as Bitcoin and Ethereum, via cross chain messaging protocols.

03

Parallel processing for non-competing transactions in a cloud-based environment.

04

Basic components for a minimum viable Block and Genesis Smart Contract Collection for each Chain, which reduces data complexity and allows for high customisation at the same time.

05

Permission for stakeholders to approve amendments to the protocol, including redefining the Consensus Protocol; Permission for SideChains to join or exit from MainChain dynamically based on the Consensus Protocol.

aelf's primary strength lies in its architecture, which utilises a multi-layered structure comprising a MainChain and multiple SideChains. This differs from a traditional Single Chain system in that aelf is a "Branched Ecosystem", where the MainChain works as the backbone of the system and connects to multiple SideChains. Compared with the traditional "one Chain to any type of Contract", aelf's "One Chain to one type of Contract" design allows aelf to distribute tasks efficiently and process transactions in parallel, significantly enhancing scalability and reducing congestion. At the same time, the SideChains are easier to manage and can be tailored for specific applications, enabling aelf to support a wide range of use cases without compromising performance.

For aelf's consensus mechanism, it utilises the Delegated Proof of Stake (DPoS) to manage overall network efficiency.

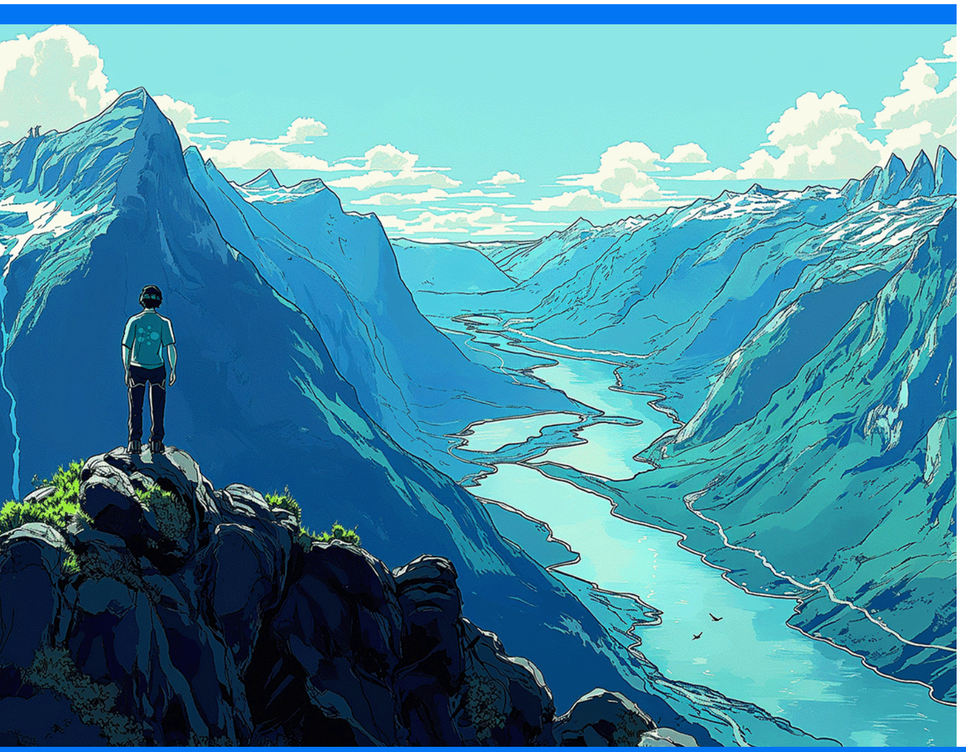
DPoS is known for its speed and energy efficiency compared to other consensus mechanisms like Proof of Work (PoW). By allowing token holders to vote for delegates who validate transactions and maintain the network, aelf ensures a high level of security and decentralisation while keeping transaction times low. Additionally, aelf's dynamic indexing and comprehensive token system offer robust support for various applications, from digital asset management to DeFi platforms, making it a versatile and powerful blockchain solution for all types of developers.

Last but not least, the C# programming language forms the foundation of the entire aelf blockchain. C# is a robust, object-oriented language with strong type-checking, which enhances code reliability and reduces runtime errors and are crucial for the immutable and secure nature of blockchain applications.

Its performance characteristics, including efficient memory management and speed, are beneficial for blockchain's demanding requirements. Further, C#'s support for asynchronous programming and parallel processing helps optimise the performance of blockchain nodes, ensuring they can handle multiple transactions simultaneously without significant latency. This strong technical foundation underpins aelf's ability to handle high transaction volumes and complex operations with ease.

For further details on aelf's original design and architecture, please kindly refer to our original whitepaper.

The evolving blockchain landscape

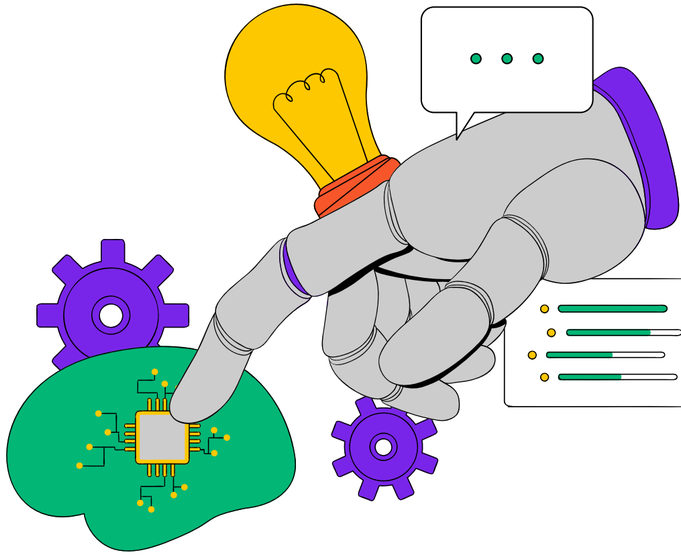


The blockchain landscape continues to evolve rapidly, with significant trends shaping the industry in 2024. DeFi remains at the forefront, characterised by its ability to eliminate intermediaries and offer direct, trustless transactions. The integration of AI-driven risk management systems enhances the security of DeFi platforms, ensuring safer transactions and compliance with regulatory requirements through automated AML and KYC processes. Additionally, cross-chain interoperability is becoming increasingly important, allowing seamless transactions and asset transfers across different blockchain networks. For instance, Chainlink's CCIP allows for cross-chain liquidity pools and interoperable DeFi protocols to leverage unique features of multiple blockchains, such as utilising Ethereum's high liquidity while benefiting from the low transaction fees of Layer 2 solutions. This interoperability is crucial for expanding DeFi services beyond the limitations of a single blockchain, thus enhancing liquidity and optimising returns for users across various protocols.

NFTs also continue to gain traction, with mainstream adoption by brands and the integration of NFTs into various sectors like gaming and real estate. The tokenisation of real-world assets (RWAs) using NFTs offers investors new opportunities by providing digital representations of physical assets, thus unlocking liquidity in traditionally illiquid markets. Ondo Finance for instance is a significant player in the tokenised treasury market, bringing institutional-grade financial products to the crypto space. They focus on tokenising U.S. Treasury Bills, offering a stable yield through their USDY stablecoin. This trend is expected to democratise access to valuable assets and promote global market participation.

Moreover, the demand for scalable and sustainable blockchain solutions is driving the adoption of energy-efficient consensus mechanisms, such as PoS, aligning with broader global sustainability initiatives. Ethereum's upgrade to Ethereum 2.0 aims to significantly enhance its scalability through the implementation of shard chains and the transition from PoW to PoS. Sharding allows the Ethereum network to be divided into smaller, more manageable pieces, which can process transactions in parallel, thus increasing overall throughput. Innovative approaches are being employed to tackle scalability issues in blockchain technology, ensuring that networks can meet growing demand without compromising on performance or security.

As blockchain technology continues to evolve and mature, its application across diverse use cases highlights its potential to enhance transparency, security, and efficiency in various sectors closer to home.



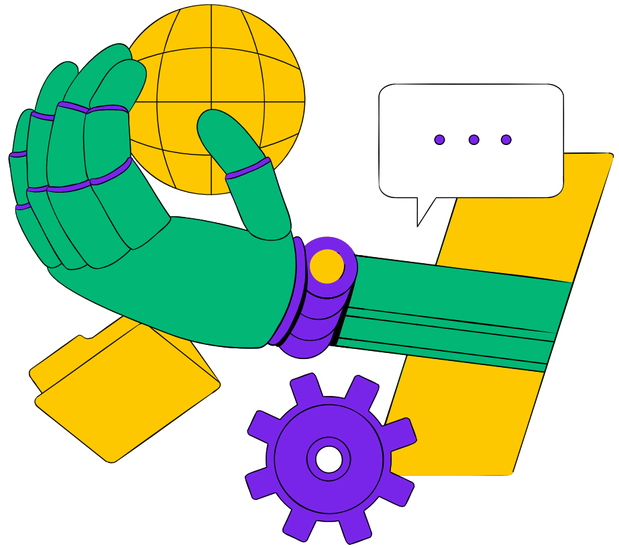
The AI revolution and its impact on blockchain

AI's influence is often subtle but profoundly transformative, reshaping our daily lives and work in remarkable ways. In healthcare, AI is making significant strides where physicians are now leveraging AI to enhance diagnostic accuracy and develop tailored treatment plans, leading to improved patient care and faster recovery times. AI is also accelerating drug discovery by analysing vast datasets to identify potential new medications, substantially reducing the time and resources typically required for extensive clinical trials. In the financial sector, financial institutions employ AI to bolster security measures such as detecting fraudulent activities by identifying unusual patterns that might elude human observation. AI is revolutionising personal finance by offering customised financial advice, helping individuals better manage their assets through analysis of their financial behaviours and providing tailored recommendations. Think Robo-advisors.

Within the blockchain sector in particular, AI offers interesting solutions to enhance security and boost efficiency. AI can predict and prevent security breaches, optimise transaction verification processes, and provide in-depth insights into on-chain activities. In smart contract development, AI can assist with code generation, optimisation, and bug detection. A blockchain's scalability and performance can be improved by AI through optimised transaction routing and network congestion prediction, resulting in transactions being processed more quickly and with fewer errors, making blockchain technology more practical for everyday use.

The blockchain's decentralised infrastructure is also opening new avenues for AI development. It offers a transparent and immutable platform for storing and sharing AI models, ensuring their integrity and facilitating collaborative development. The concept of decentralised AI marketplaces such as SingularityNet, enabled by smart contracts, could allow for the buying, selling, and execution of AI services in a trustless environment. Furthermore, native cryptocurrencies on blockchains can be utilised to incentivisee AI training, potentially democratising access to advanced AI capabilities. In terms of user experience in blockchain applications, AI can offer personalized recommendations on DeFi platforms, aiding users in making more informed investment decisions, or assist in managing digital identities, ensuring data security while simplifying user access.

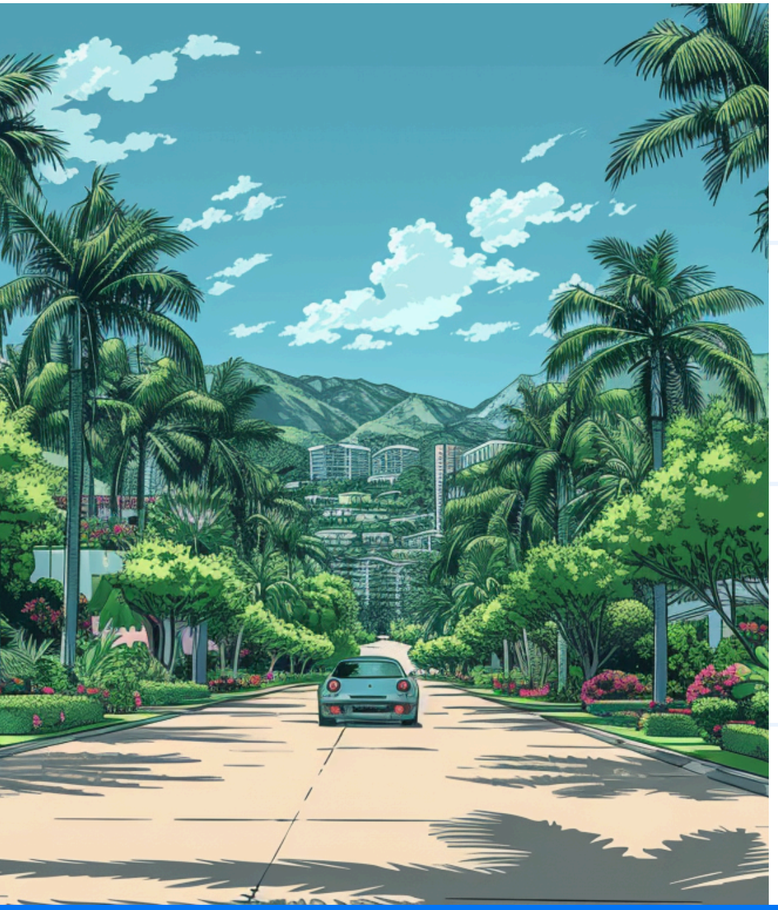
As we continue to find synergy between AI and blockchain, we earnestly embrace further innovative developments here at aelf and anticipate broader adoption of blockchain technology into our daily lives. This collaboration is not just advancing both technologies but is also delivering tangible benefits to users in the form of security, scalability and user experience of the entire Web3 ecosystem.



aelf’s AI-enhanced architecture

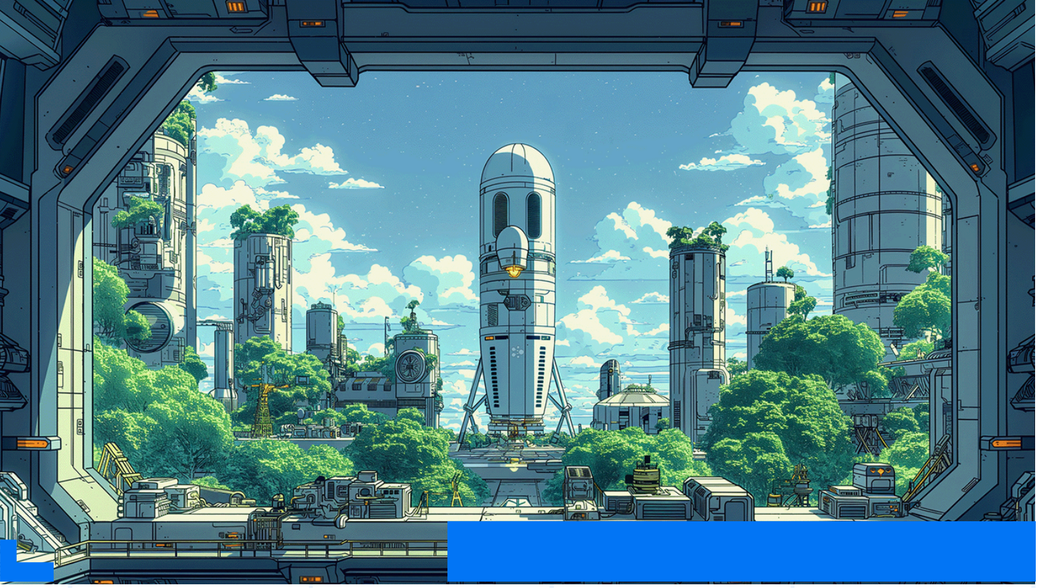
As mentioned earlier, aelf's underlying architecture is fundamentally written in C#. C# offers a powerful foundation for AI integration in blockchain development, primarily due to its alignment with the .NET ecosystem, which provides a wide range of libraries and tools that facilitate the development of sophisticated AI models, including the integration of machine learning algorithms, natural language processing (NLP), and data analytics. These capabilities are crucial for enhancing the functionality and intelligence of blockchain platforms.

C#’s strong typing and object-oriented features ensure that code is reliable and maintainable, reducing the likelihood of errors that could compromise security. Furthermore, the seamless integration with cloud computing platforms such as Google Cloud featuring AI and machine learning services allows aelf developers to easily implement AI functionalities within the aelf blockchain. This includes capabilities such as automated smart contract execution, fraud detection, and enhanced data privacy through advanced encryption techniques.



By leveraging C# and .NET's extensive libraries, aelf developers can incorporate AI models that predict network congestion and dynamically adjust the consensus process to maintain high performance. This not only ensures efficient network management but also enhances user experience by reducing transaction latency and costs.

For developers looking to build on aelf, they can also leverage AI for various functionalities with the extensive support from popular professional development tools, which further streamlines the development process, allowing for rapid prototyping and deployment of AI-enhanced blockchain solutions such as financial dApps and decentralised marketplaces. Overall, C# provides a robust base for AI integration in blockchain platforms like aelf by leveraging the strengths of the .NET ecosystem including extensive libraries, seamless integration with best-of-class cloud computing AI services, and powerful development tools, all of which contribute to creating secure, scalable, and intelligent blockchain applications.



AI integration points in the aelf ecosystem

The integration of AI into the aelf ecosystem marks a significant advancement in the blockchain's operational capabilities which not only optimises performance and security, but also plays a crucial role in transitioning mass Web2 users to Web3 by significantly improving the overall user and developer experience.

The integration of AI into the aelf ecosystem can be broadly categorised into two main areas, which we will detail below:

01

Operational Excellence

02

Developer and user experience

Operational Excellence

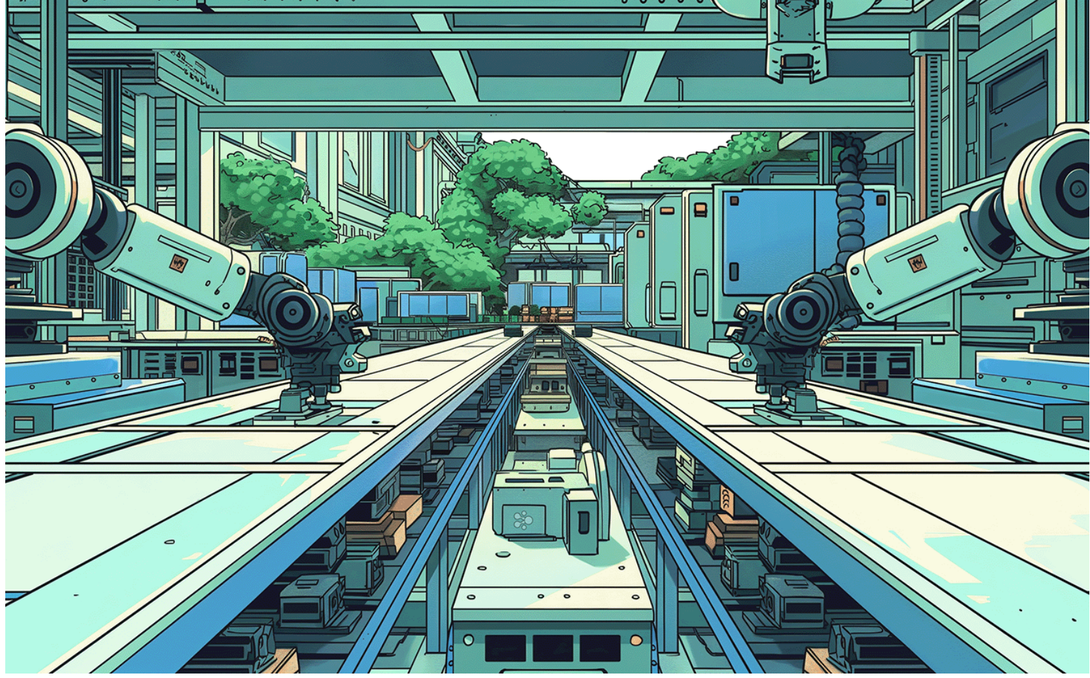
01
Machine Learning Models for Smart Contract Optimisation

Integrating machine learning models into the aelf blockchain can significantly optimise gas usage and execution efficiency. We draw inspiration from existing rollup technologies but enhance them through advanced predictive modeling.

Optimising Gas Usage

Machine learning models can analyse historical data on smart contract executions to identify patterns and predict the optimal gas prices for future transactions. This approach is similar to the predictive models used in financial markets to forecast asset prices and trading volumes, ensuring that transactions on aelf are both cost-effective and timely. Another practical application for machine learning models is the detection and minimisation of gas-expensive patterns in smart contract code. aelf has adapted tools that use machine learning algorithms to analyse and identify inefficient code patterns that consume excessive gas. For example, replacing arrays with mappings where iteration is not required can significantly reduce gas costs, as mappings are generally less expensive in terms of gas usage. By continuously monitoring and optimising these patterns, aelf can ensure that smart contracts are executed with minimal gas expenditure, similar to optimisation techniques employed by developers on the Ethereum network.

Moreover, machine learning can optimise the execution efficiency of smart contracts by dynamically adjusting resource allocation based on the predicted computational requirements. Techniques such as variable packing, which arranges variables to fit into fewer storage slots, can be automatically identified and applied by machine learning algorithms. This reduces the number of storage operations needed, thereby lowering gas consumption. Additionally, employing machine learning on aelf to manage on-chain and off-chain data storage decisions can optimise the use of memory and storage, further enhancing gas efficiency.



Enhancing Execution Efficiency

Beyond gas optimisation, AI can also streamline the execution efficiency of smart contracts on aelf. By employing predictive modeling, AI can anticipate resource requirements and execution paths for complex smart contracts. This foresight allows the system to allocate computational resources more effectively, ensuring that contracts run smoothly without unnecessary delays.

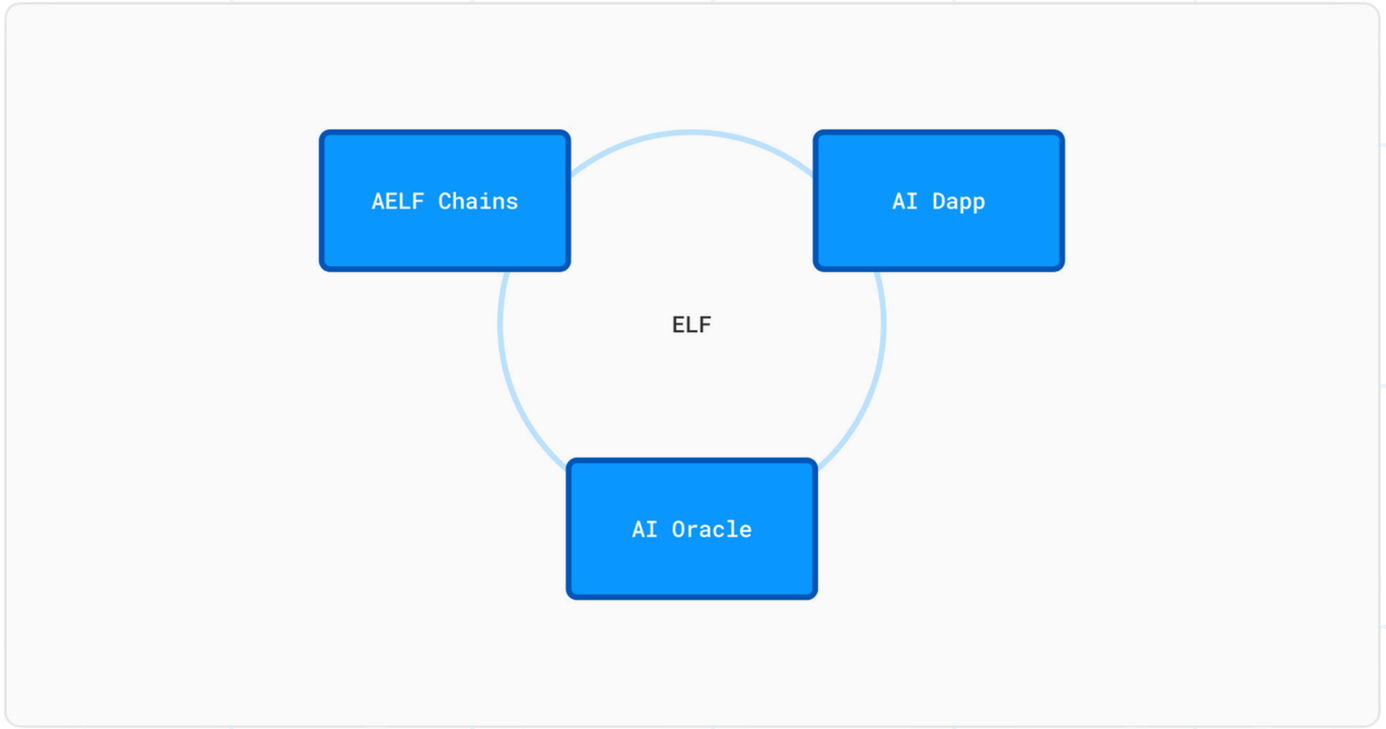
This can be seen today in how Google Cloud's AI optimises resource allocation for its cloud services. By using machine learning models to predict demand, Google Cloud can dynamically adjust resource allocation to meet user needs efficiently. In the case of aelf, this involves using AI to forecast transaction volumes and adjust computational resources in real-time. During a DeFi surge for instance, AI models can predict increased transaction loads and preemptively allocate more computational power to handle the surge without network slowdowns.

Predictive models can evaluate the execution patterns of different smart contracts and suggest optimisations in their code to reduce execution time and resource consumption. This is similar to the optimisation techniques used in compiler design, where code is analysed and restructured to improve performance. Another real-life example of this concept is the work done by the Ethereum Foundation to optimise the EVM. Researchers have used AI to analyse smart contract bytecode, detect inefficient loops or redundant calculations and suggest optimisations that reduce gas usage and execution time. By applying similar techniques, aelf can enhance the efficiency of its smart contracts.

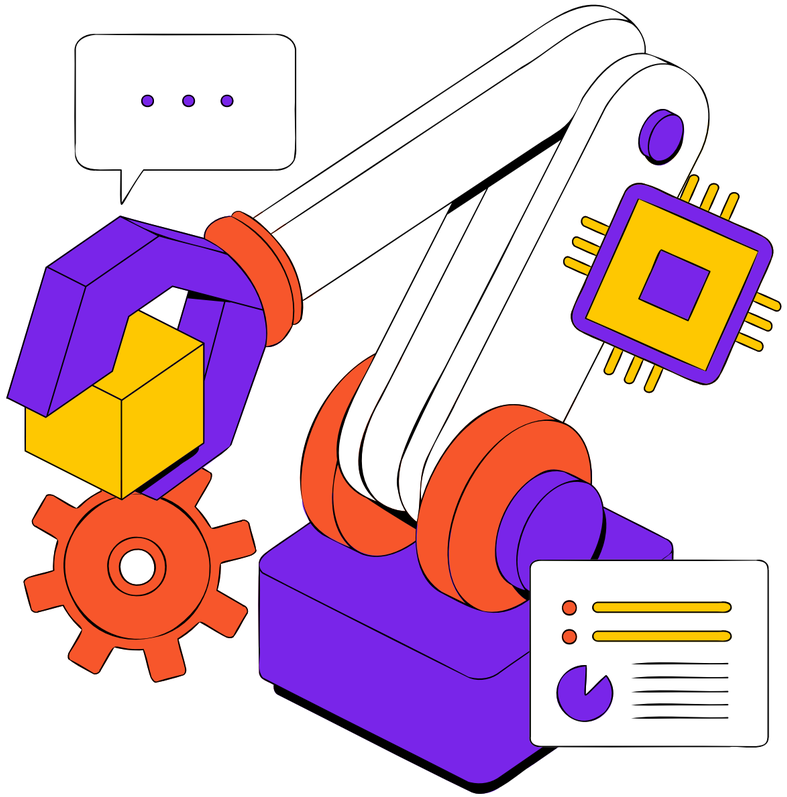
Operational Excellence

02
AI Oracle on aelf

aelf's AI oracle positions itself as the AI layer of Web3, leveraging both oracle and security compute technology to facilitate the development of AI-powered dApps. It plays a crucial role in aelf's architecture, which comprises the aelf chains, the AI oracle network, and AI applications, serving as the bridge and hub for integrating AI capabilities into the blockchain.



02
AI Oracle on aelf

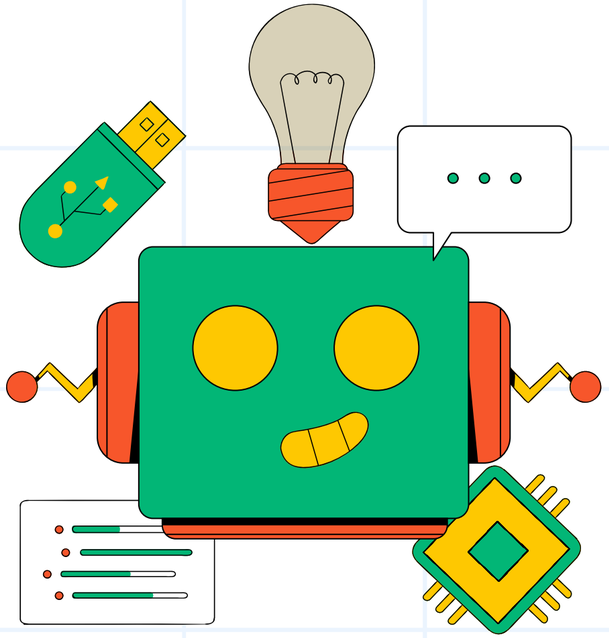


On-chain AI

Currently, the AI oracle field primarily focuses on ensuring the integrity of the AI computation process (such as ORA) by incorporating Zero-Knowledge Proofs (ZKP) at each step of the calculation and reasoning process. This approach ensures that the reasoning process is derived from the Large Language Model (LLM) itself, rather than human intervention. However, merely guaranteeing the integrity of the AI reasoning process is insufficient. It is also crucial to ensure that the higher-layer AI Agent application remains tamper-proof.

Additionally, widely used large models such as ChatGPT or decentralised computing networks like IO have their own computation and operation mechanisms, making it impractical to embed the ZKP process. Therefore, we believe that ensuring the integrity of the APIs called by the large model and the upper-layer AI Agent application is more critical and valuable than merely focusing on the AI inference process.

02
AI Oracle on aelf



Implementation on aelf

aelf employs the following mechanisms to implement the AI oracle network:

TLSNotary Mechanism
This ensures that the API provided by the underlying centralised or decentralised LLM remains untampered from the data source to the local endpoint.

SGX and Decentralised Computing Networks
These technologies guarantee the trustworthy operation of AI agents.

The aelf oracle architecture is primarily divided into two components: **1. on-chain** and **2. off-chain**.

1. On-Chain Components

The on-chain components (currently on the aelf chain, with potential future deployment on Ethereum or other chains) include node pledge management, a reputation system, fee and income settlement, an AI Agent scheduler, and an AI Agent factory contract. These components form the foundational framework contracts for the AI oracle. Combined with off-chain nodes, they constitute the aelf oracle network. Specifically, the developer-defined AI oracle contract is generated by the AI Agent factory contract based on user-defined logic.

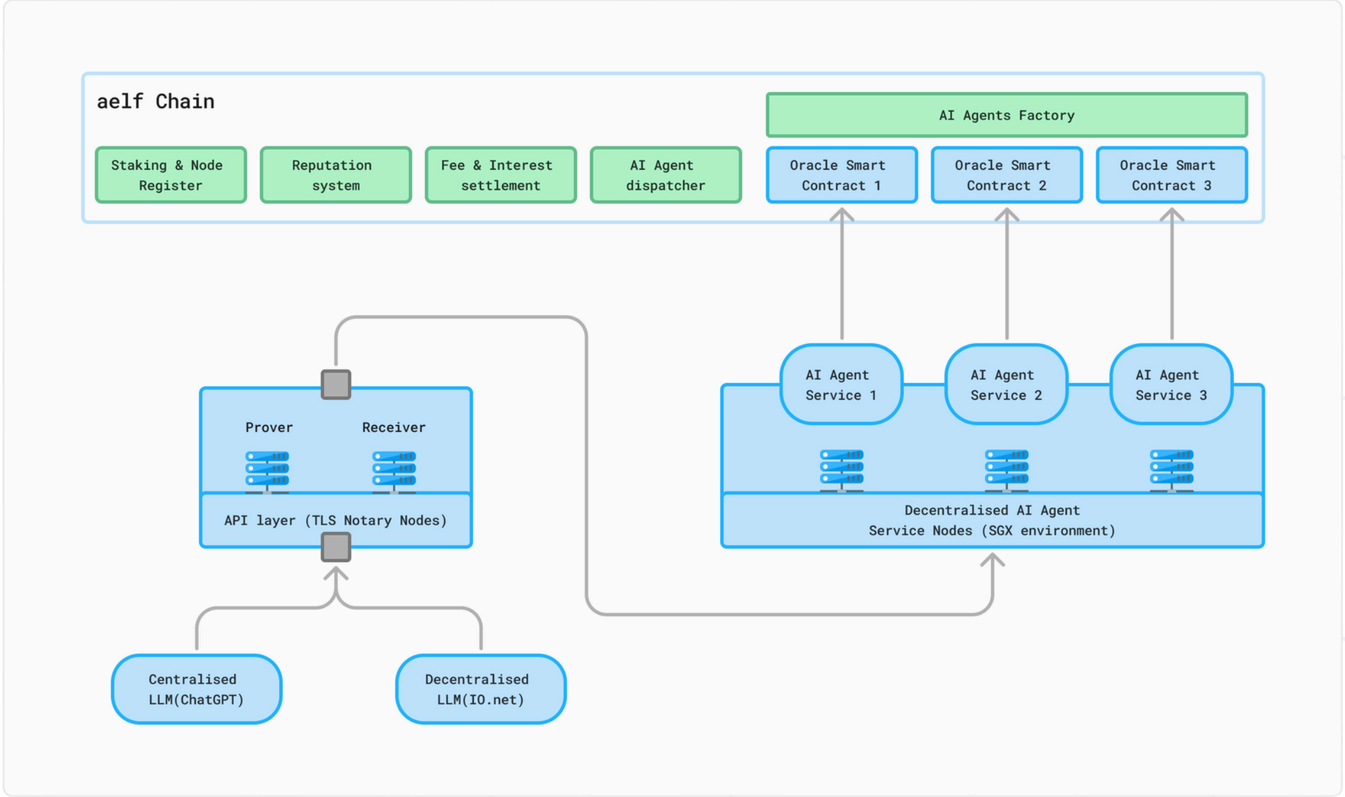
Node Pledge Management: Manages the AI Agent Service Nodes. Any node joining the network must pledge a certain amount of ELF.

Reputation System: Evaluates node operating status in conjunction with the committee's assessments, such as downgrading offline nodes.

Cost and Revenue Settlement: Integrates with the AI Agent scheduler to settle revenue for nodes and charge fees for dApp calls.

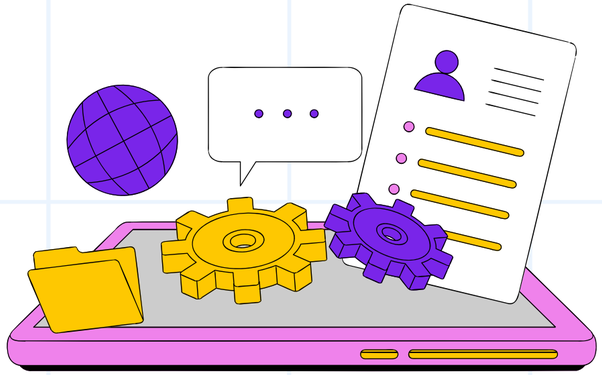
AI Agent Scheduler: Deploys and schedules AI Agent Service code based on network computing power load.

AI Agent Factory: Generates custom contracts for any user-defined AI Agent logic. It acts as the intermediary between the foundational framework contract and the user-defined contract, forwarding the dApp call interface, and interacting with the fee and income settlement contract. Additionally, it mediates between the AI Agent Service and user-defined contracts, processing all calls and inference results of the AI Agent Services uniformly.



2. Off-Chain Components

The off-chain components consist of TLS Notary Nodes and Decentralised AI Agent Service Nodes (SGX environment).



TLS Notary Nodes

TLS Notary is an attestation mechanism that ensures the integrity and security of network communications between untrusted participants using the TLS (Transport Layer Security) protocol. This protocol encrypts communications between clients and servers. The "Notary" refers to a trusted third party or node that verifies the integrity of the TLS session between the client and server, ensuring it has not been tampered with. TLS Notary captures and records key information from the TLS session, such as the encrypted handshake and session keys, enabling even untrusted participants to verify the session's integrity. This mechanism allows the upper-layer AI Agent Service to authenticate data from external servers without fully trusting them. Evidence from TLS Notary can be used in the on-chain verification process to ensure the AI Agent Service processes safe and trustworthy data. aelf currently employs TLS Notary Nodes through two independent nodes: Receiver and Notary.

Decentralised AI Agent Service Nodes

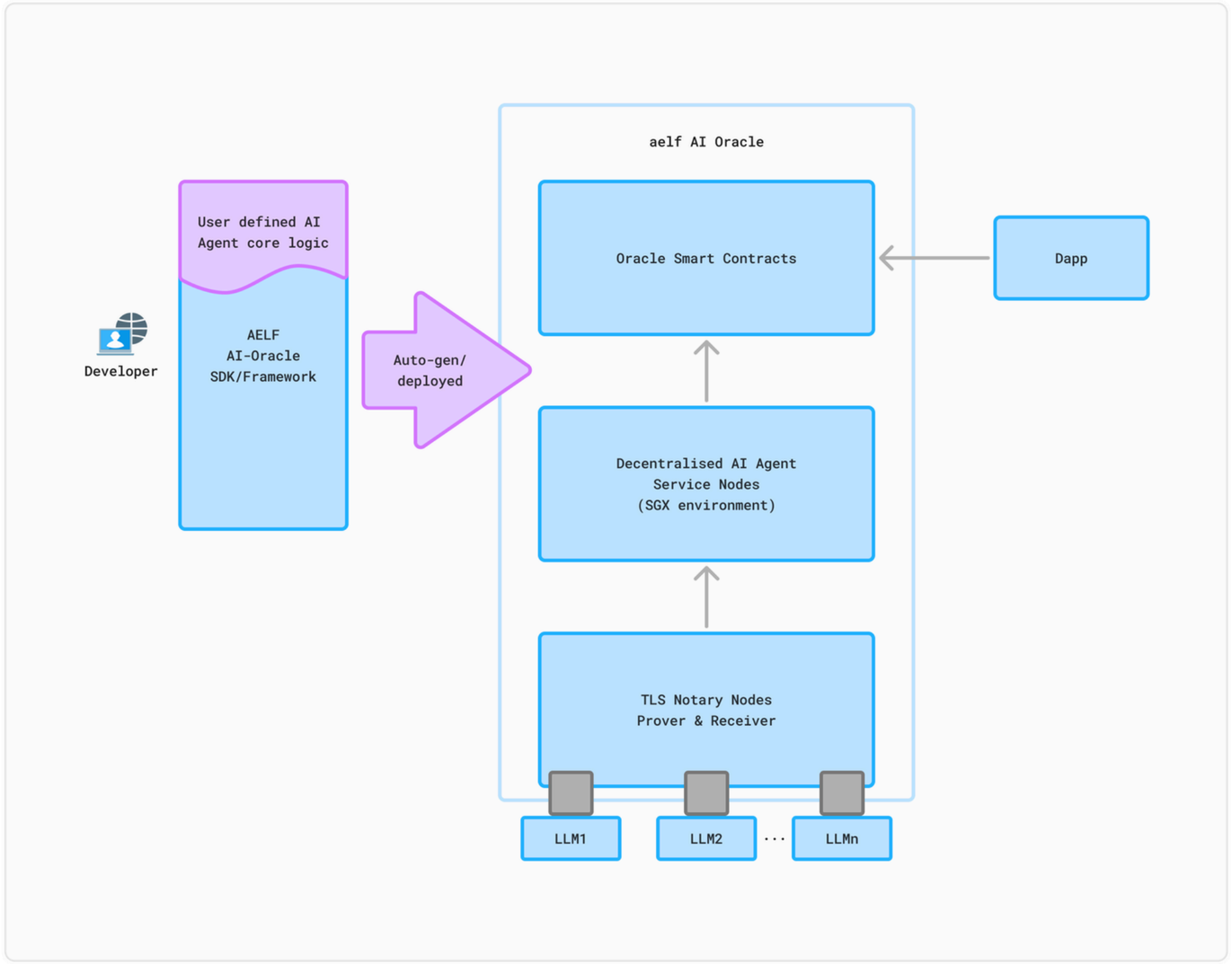
The Decentralised AI Agent Service Nodes form a decentralised AI Agent computing network utilising Intel's SGX (Software Guard Extensions) technology. SGX creates a "trusted execution environment" at the processor level, providing secure and reliable data processing capabilities. In the aelf oracle network, the AI Agent Service operates within the SGX execution environment, ensuring that even node operators cannot access the details of the AI Agent Service execution. This SGX-based technology protects the security of AI Agent code operations from tampering, and when combined with blockchain technology, provides tamper-proof records and decentralised trust. Developers deploy and run their AI Agent Services on this secure layer of the aelf oracle network.

AI Oracle Use on aelf

aelf will provide a comprehensive SDK and framework and developers only need to develop custom core logic and configure the interaction interfaces. The framework can automatically generate, schedule, and deploy the relevant AI Agent Services to the oracle network, and trigger the AI Agent Factory to generate the associated AI smart contracts.

For simple AI call logic, this process is fully automated. However, for more complex AI Agent Services (the framework currently integrates with the ChatGPT API and can connect with additional LLMs in the future), such as memory and embeddings, developers will need to develop these services themselves using the provided interface documentation.

Additionally, the aelf AI Oracle offers traditional services such as price feeds and random number generation. Using the aelf AI Oracle SDK, developers can customise their code to create limitless on-chain AI application scenarios.



The aelf AI Oracle is committed to fostering the development of the AI Dapp ecosystem on aelf. This involves optimising the SDK and framework to enhance the interaction capabilities between AI Dapps and AI, as well as strengthening collaborations with AI Agent platforms. Currently, aelf AI Oracle will integrate with the ChatGPT API, and in the future, it plans to incorporate more centralised LLMs and collaborate with other decentralised AI computing networks to meet the growing needs of developers.

Beyond the aelf chain, aelf AI Oracle aims to expand to other major blockchains such as Ethereum, truly becoming the AI layer of Web3.

Operational Excellence

03
Intelligent Load Balancing for aelf

Intelligent load balancing is a critical component for enhancing the scalability and performance of the aelf blockchain. By distributing the workload across the network more effectively, AI and machine learning can ensure that no single node becomes a bottleneck, thereby maintaining high transaction throughput and reducing latency.

Dynamic Traffic Distribution

Machine learning algorithms can analyse real-time network traffic and historical transaction patterns to dynamically distribute the workload. An AI model can predict which nodes are likely to become overloaded based on their current and past performance. By redirecting traffic away from these nodes to less utilised ones, the aelf network can prevent congestion and maintain smooth operations. A practical example of this outside of blockchain can be seen in content delivery networks such as Akamai and Cloudflare, which use AI to manage traffic distribution across global server networks. These systems continuously monitor server load, latency, and other metrics to ensure that content is delivered quickly and efficiently to users. Similarly, aelf intends to implement AI-driven load balancing to optimise transaction processing across its nodes.

Resource utilisation optimisation

AI can also optimise the use of network resources by balancing the computational load. This involves distributing smart contract execution and transaction validation tasks evenly across the network. Machine learning models can identify nodes with excess computational capacity and redirect tasks to these nodes, ensuring that all resources are utilised effectively. For example, Google's Borg system, which manages resources across its massive data centers, uses machine learning to allocate tasks dynamically based on available resources. This ensures high utilisation rates and minimises idle time. aelf is currently evaluating similar strategies to balance the workload across its network, ensuring that no single node is underutilised or overwhelmed.

Predictive Scaling

Predictive scaling is an important aspect of intelligent load balancing where AI models forecast future demand and scale resources accordingly. By analysing trends and patterns in transaction volumes, AI can predict when the network will experience high traffic and preemptively allocate additional resources to handle the load. This proactive approach ensures that the network remains responsive even during peak times. AWS Auto Scaling uses predictive algorithms to automatically adjust the number of active servers based on predicted load. This approach minimises response times and prevents service degradation. aelf leveraging similar predictive scaling techniques to ensure that its network can handle fluctuations in transaction volume without compromising performance.

Fault Tolerance and Redundancy

Intelligent load balancing also enhances fault tolerance and redundancy within the network. AI can monitor node health and performance, identifying and isolating nodes that exhibit signs of failure or degraded performance. By redirecting traffic away from these nodes and distributing it among healthy nodes, the network can maintain high availability and reliability. Systems like Netflix's Chaos Monkey, which intentionally induces failures in their infrastructure to test resilience, rely on intelligent load balancing to reroute traffic and maintain service continuity. aelf will be implementing similar AI-driven fault tolerance mechanisms to ensure that its network remains robust even in the face of individual node failures.

Intelligent load balancing, powered by AI and machine learning, offers substantial benefits for enhancing the scalability and performance of the aelf blockchain. By dynamically distributing traffic, optimising resource utilisation, enabling predictive scaling, and enhancing fault tolerance, AI-driven load balancing ensures that the aelf network can handle increasing transaction volumes efficiently.

Operational Excellence

04
AI-Facilitated Cross-Chain Interoperability for aelf

Incorporating AI into aelf's cross-chain interoperability framework can significantly enhance the efficiency and security of data transfer and validation processes. By leveraging AI-driven optimisations, aelf can ensure seamless communication and robust performance within itself between MainChain and SideChains, and more importantly with various other blockchain networks.

Optimising Data Transfer

AI will be instrumental in streamlining the data transfer process between the various blockchains connected to aelf. Machine learning algorithms can analyse historical data and current network conditions to predict optimal routing paths for data packets. This ensures that data is routed through the least congested and most efficient paths, reducing latency and improving transfer speeds. For instance, AI models can dynamically adjust routing paths based on real-time network traffic, similar to how AI optimises data flow in large-scale cloud computing networks used by companies like Google and Amazon. Furthermore, AI can employ predictive analytics to forecast network congestion and preemptively adjust data transfer protocols. This capability allows aelf to avoid potential bottlenecks and ensure that cross-chain transactions are processed efficiently. By continuously learning from network behavior, AI algorithms can improve their predictions over time, leading to progressively better performance and reliability.

Enhanced Data Validation

AI can also enhance the security and accuracy of cross-chain transaction validation. Machine learning models, trained on vast datasets of transaction histories, can identify patterns and anomalies that may indicate fraudulent activity or errors. AI-driven tools can flag transactions that deviate from typical behaviour, such as unusually large transfers or repeated transactions from the same node, which could indicate a security breach or double-spending attempt. aelf is working with several potential strategic partners to implement similar AI-driven validation mechanisms to ensure the integrity and security of cross-chain interactions.

Operational Excellence

05
AI-Driven Node Selection and Validation for aelf

As part of a longer-term initiative, our research team is currently in the exploratory phase of evaluating how AI might be incorporated into aelf's node selection and validation processes. This initiative, if pursued, can potentially significantly enhance both the speed and security of aelf consensus mechanisms.

Optimising Node Selection

Machine learning can be employed to analyse historical data and predict which nodes are most likely to perform reliably and securely. By evaluating factors such as past performance, network latency, and resource availability, AI models can prioritise nodes that are best suited for participating in the consensus process. For example, a neural network could be trained to predict node reliability based on metrics like uptime, transaction validation speed, and historical accuracy, ensuring that only the most capable nodes are selected for consensus.

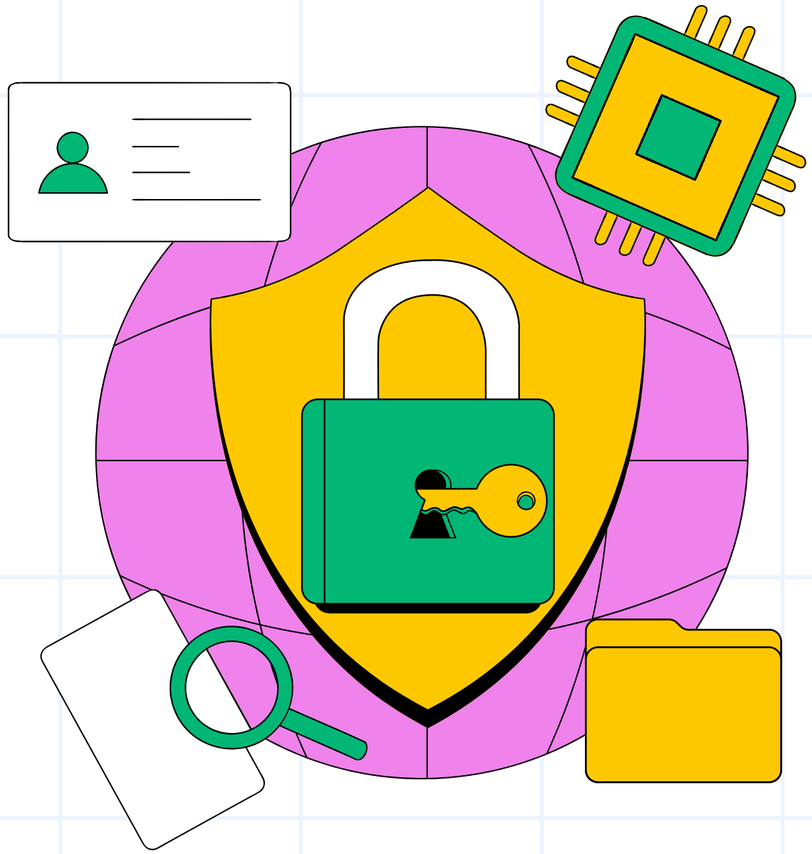
One practical approach to achieving this involves the use of genetic algorithms, which are inspired by the process of natural selection. These algorithms can evolve optimal strategies for node selection by continuously iterating and improving based on predefined fitness criteria. For example, nodes that have demonstrated high reliability and low latency in past transactions are given higher priority, ensuring that the consensus process remains efficient and secure. This method has been effectively utilised in other domains, such as edge computing and task scheduling, to optimise resource allocation and performance.

Enhancing Consensus Security

AI can also enhance the security of the consensus process by dynamically adjusting the selection criteria based on real-time network conditions and potential threats. For instance, machine learning models can detect unusual patterns that might indicate a security threat, such as a potential Sybil attack where multiple nodes controlled by a single entity attempt to dominate the network. By identifying and mitigating these risks in real-time, AI ensures that the consensus mechanism remains robust and secure.

A specific implementation of this concept could involve anomaly detection algorithms. These algorithms can monitor network activity for deviations from normal behaviour, such as sudden spikes in node participation or unusual transaction patterns. When such anomalies are detected, the AI system can flag these nodes for further scrutiny or exclude them from the consensus process, thus protecting the network from potential attacks. This approach leverages AI's ability to process and analyse large volumes of data in real-time, providing a proactive security measure for the blockchain.

As node elections are fundamental to network consensus and security, we may integrate the aelf AI Oracle to ensure that the elections are open, transparent, and trustworthy.



Developer and User Experience

01
Enabling Non-Technical Users to Create Smart Contracts with NLP on aelf

The complexity of creating smart contracts in aelf, presents significant challenges for non-technical users or even technical users who may not be familiar with the Web3 environment. Users often struggle with understanding programming languages, navigating development environments, and ensuring the security and functionality of their contracts. By leveraging Natural Language Processing (NLP), aelf can simplify this process, allowing users to create end to end smart contracts through intuitive natural language inputs.

Simplifying Smart Contract Creation

NLP can transform the way non-technical users interact with blockchain technology by interpreting natural language descriptions and converting them into executable smart contract code. Users can describe their contractual needs in plain language, such as "Create an escrow contract where Alice deposits 10 ELF, and Bob receives it upon satisfying X condition" The NLP model can then interpret this input, extract relevant entities and conditions, and generate the corresponding smart contract code. This process leverages the capabilities of advanced NLP models like GPT-4 and OpenAI's Codex, which are trained to understand and generate code based on natural language descriptions. These models can map user inputs to predefined templates or dynamically generate code structures that match the user's requirements. This allows non-technical users to create complex smart contracts without needing to write a single line of code.

Enhancing Accessibility and Usability

By integrating NLP into its platform, aelf intends to significantly enhance the accessibility and usability of blockchain technology. This democratisation of technology enables a broader audience to participate in the blockchain ecosystem, driving adoption and innovation. For example, consider a user who wants to set up a simple DeFi contract for staking tokens. Instead of hiring a developer, the owner can use an NLP-driven interface to describe the terms of the contract. The system interprets these terms and generates a smart contract that handles the various staking functionalities, ensuring that the DeFi protocol works as intended and most importantly, secure.

Real-Time Feedback and Validation

Another key benefit of using NLP for smart contract creation on aelf is the provision of real-time feedback and validation. As users input their requirements, the NLP system can highlight potential issues, suggest optimisations, and explain complex logic in simple terms. This interactive process ensures that users understand the generated code and can make informed decisions about their contracts. For instance, if a user inputs a contract description that could lead to a security vulnerability, the NLP system can flag this and provide suggestions for improvement. This not only enhances the security of the contracts but also educates users about best practices in smart contract development.

Broadening the User Base

Implementing NLP for smart contract creation allows aelf to broaden its user base by catering to individuals and businesses without technical backgrounds. This inclusivity is crucial for the widespread adoption of blockchain technology. By lowering the barriers to entry and simplifying the process, aelf intends to attract a diverse range of users who can benefit from the automation and security of smart contracts without needing to become blockchain experts.

Developer and User Experience

02
AI based Smart Contract Audit for Developers on aelf

AI auditing of smart contracts leverages large-scale pre-trained models, such as GPT-4 and BERT, to automate the audit process. Unlike traditional audits, which rely on manual code reviews and static analysis tools, AI auditing utilises natural language processing (NLP) and deep learning to deliver more comprehensive and accurate audit reports.

In AI auditing, smart contract code is input into a pre-trained model, which analyses the code's syntax and logical structure to identify security vulnerabilities and logical errors. Users interact with the model using prompts to guide the analysis, such as "Check for reentrancy attack vulnerabilities in this smart contract." The model then performs a detailed analysis and generates an audit report.

The audit report details potential vulnerabilities and provides corresponding remediation suggestions.

Key Aspects of AI based Smart Contract Audits

The primary goal is to review and analyse the code before deployment to identify vulnerabilities, inefficiencies, or potential leaks. This process is critical, as defects can lead to significant financial losses or security breaches. For instance, audits of DeFi application contracts rigorously test for exploits to protect user funds.

Vulnerability Assessment:
Identifying potential security issues in the code. For example, an audit of a token issuance contract uncovered a vulnerability that could allow fund theft, enabling developers to resolve the issue pre-launch.

Code Optimisation:
Improving efficiency and reducing gas fees. Auditors recommended optimisations for a game DApp's contract, significantly lowering transaction fees and enhancing accessibility and affordability for players.

Compliance Check:
Ensuring adherence to legal and regulatory standards. Audits verify that smart contracts meet new regulatory requirements, ensuring DeFi platforms remain compliant.

Technical Implementation on aelf

This process typically involves two approaches: prompt design and fine-tuning.

1. Prompt Design Approach

The prompt design approach involves constructing specific prompts to guide the model to perform optimally in certain tasks without requiring retraining. In the context of smart contract auditing, the steps are as follows:

1 Define the Target Task

Clearly outline the specific issues to be audited and establish an audit rule library. For example:

- Reentrancy Attack Vulnerability: Determine if the contract allows external contracts to repeatedly call, leading to multiple fund transfers.

2 Design Prompts

Create precise prompts to guide the model's analysis. For example:

- "Check if this smart contract has reentrancy attack vulnerabilities."

3 Input Prompts and Contract Code

Enter the designed prompts and smart contract code into a large model, such as GPT-4. The steps include:

- Select a model that supports code analysis.
- Input the prompts specifying the exact issues to be checked.

4 Generate Audit Report

The model analyses the input prompts and contract code, generating an audit report that typically includes:

- **Vulnerability Description**: Detailed description of discovered vulnerabilities, including the exact location and trigger conditions.
- **Risk Assessment**: Evaluation of the severity and potential impact of the vulnerability.
- **Remediation Suggestions**: Specific recommendations for fixing the vulnerability, such as modifying code logic, adding access controls, or using secure libraries.

This method's advantage is that it does not require extensive data and computational resources, allowing for quick deployment and application on aelf.

Technical Implementation on aelf

This process typically involves two approaches: prompt design and fine-tuning.

2. Automated Fine-Tuning Approach

The fine-tuning approach involves retraining the large model to better adapt it to specific tasks. This requires a significant amount of historical audit data and remediation cases to improve the model's auditing capabilities. The steps include:

1. **Data Collection**
   Gather a large dataset of smart contract codes, audit reports, and remediation cases.

2. **Data Preprocessing:**
   Clean and format the data to ensure effective learning.

3. **Model Fine-Tuning**
   Train the large model on the preprocessed dataset to enhance its understanding and auditing capabilities.

4. **Validation and optimisation**
   Evaluate the model's performance using a validation set and make necessary optimisations.

5. **Model Deployment**
   Deploy the fine-tuned model in the actual audit environment.

This method can significantly enhance the model's performance in specific tasks but requires substantial computational resources and time. The fine-tuned model can more accurately identify complex vulnerabilities in smart contracts and provide targeted remediation suggestions.

Ultimately, this model will be integrated into a smart contract auditing tool, enabling both aelf and dApp developers to utilise it for thorough audits prior to deploying their smart contracts on the aelf chain.

Developer and User Experience

03
Enhancing User and Developer Experience with AI-Powered Chatbots on aelf

Integrating AI-powered chatbots into the aelf blockchain ecosystem can significantly enhance both user and developer experiences by providing real-time support, automating routine tasks, and facilitating access to information.

User Experience Improvement

AI-powered chatbots can dramatically improve user experience by offering instant support and personalised assistance. For instance, users new to the aelf ecosystem often have questions about setting up wallets, conducting transactions, or understanding blockchain features. aelf has partnered with an industry leading strategic partner, Chaingpt, to develop an AI chatbot that can provide 24/7 support, answering frequently asked questions, guiding users through processes step-by-step, and troubleshooting issues in real-time. This immediate assistance reduces frustration and enhances user satisfaction. This is commonplace in traditional businesses such as major banks like Bank of America that have implemented AI chatbots like Erica to assist customers with everyday banking needs, providing transaction histories, sending alerts, and offering financial advice. Similarly, an AI chatbot on aelf can help users navigate the platform more efficiently, providing information about token balances, transaction statuses, and smart contract functionalities.

AI chatbots can also enhance personalised user engagement on the aelf platform. By analysing user behaviour and preferences, chatbots can offer tailored recommendations, such as suggesting new dApps or features that might interest the user. This personalised interaction can increase user engagement and retention, making the platform more attractive to both new and existing users. This is commonly done in e-commerce platforms like Amazon that use AI chatbots to recommend products based on user browsing history and past purchases. Applying this to aelf, the chatbot could recommend specific dApps or new blockchain features based on the user's activity, thus enhancing the user's interaction with the platform

Developer Experience Enhancement

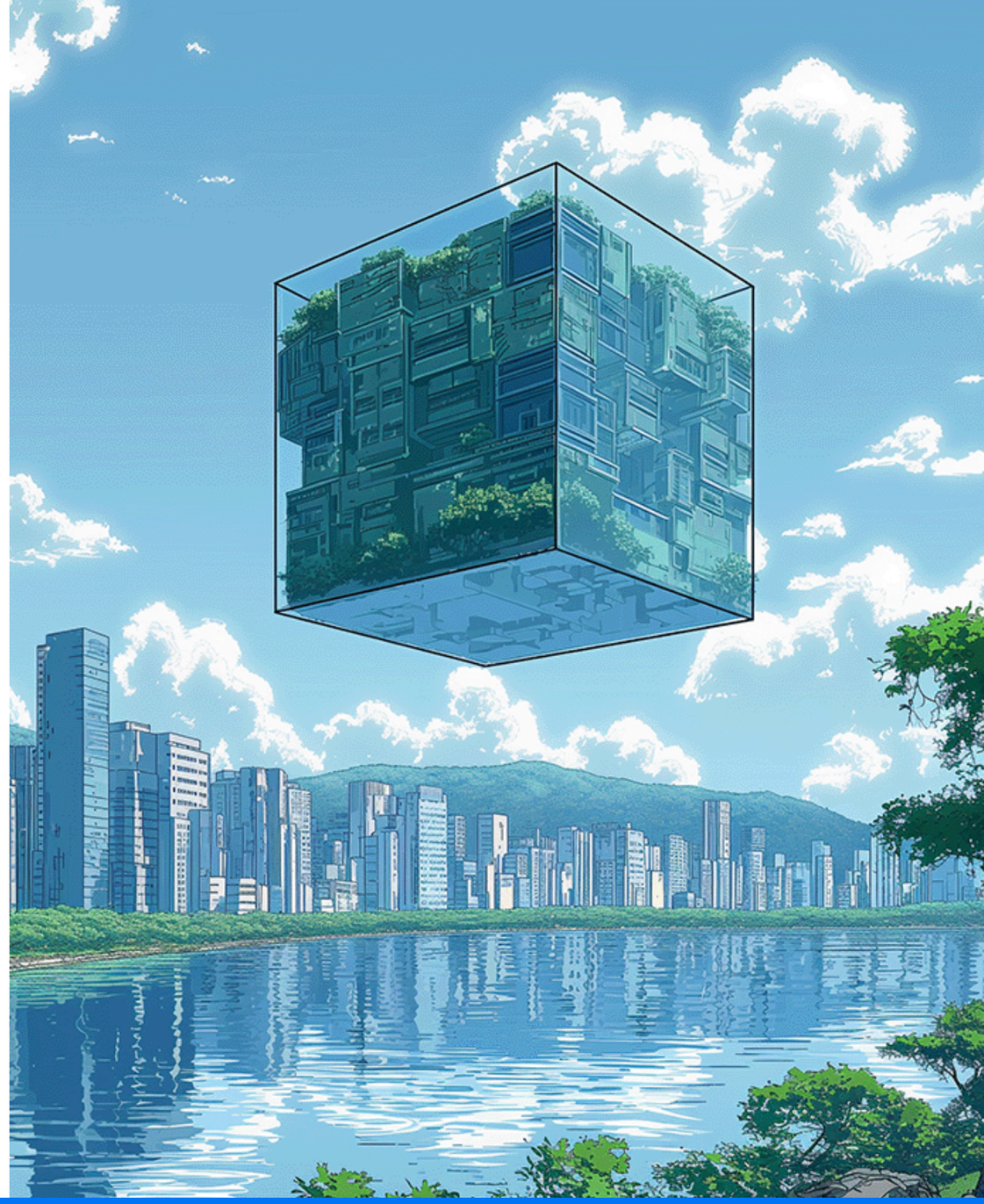
For developers, AI-powered chatbots can streamline the development process by automating repetitive tasks and offering code assistance. aelf is developing integrated chatbots that provide real-time coding help, debug suggestions, and documentation access. For instance, when a developer encounters an error while writing smart contracts in C#, the chatbot can instantly suggest solutions, recommend best practices, and provide snippets from relevant documentation. GitHub Copilot, an AI-powered code completion tool developed by OpenAI and GitHub, serves as an excellent example. It assists developers by suggesting code lines and completing functions based on the context. By integrating similar AI capabilities, aelf can make the development process more efficient, helping developers quickly resolve issues and focus on creating innovative applications. We want our developers to spend less time on coding, and more time on creating value.

Efficient Onboarding and Education

AI chatbots can facilitate efficient onboarding and continuous education for both users and developers. New users can be guided through the initial setup, learning about key features and functionalities through interactive conversations with the chatbot. For developers, the chatbot can provide tutorials, best practices, and updates on new development tools and protocols within the aelf ecosystem. Educational institutions and online learning platforms like Coursera use AI chatbots to guide students through course selections, answer questions, and provide feedback. Similarly, aelf's chatbot can act as a mentor, ensuring users and developers have a smooth and informative onboarding experience, reducing the learning curve associated with blockchain technology.

aelf as a modular blockchain

A modular blockchain is a design approach that focuses on handling a limited set of responsibilities while outsourcing the rest to one or more independent layers. It decomposes the various tasks or functions of a blockchain into distinct layers or modules. This segregation allows for improved performance, scalability, and customisation.



Execution Layer

This layer is where transactions and state changes occur, such as modifying wallet balances. Users interact with this layer through transaction operations, currency transfers, and smart contract deployments.

Settlement Layer

An optional layer in Modular Blockchain, it is responsible for verifying and settling rollup transactions. Rollup transactions are a technique that bundles multiple transactions into a single one for improved efficiency.

Consensus Layer

This layer provides ordering and finality through a network of full nodes. These full nodes download and execute block contents, reaching a consensus on the validity of state transitions.

Data Availability Layer

This layer contains data that has already been confirmed as valid. It ensures that the data required for verifying state transitions is available and easily verifiable.

By separating these tasks, a modular blockchain enables independent optimisation of each layer, leading to significant improvements across layers. This results in a more scalable, composable, and decentralised system.

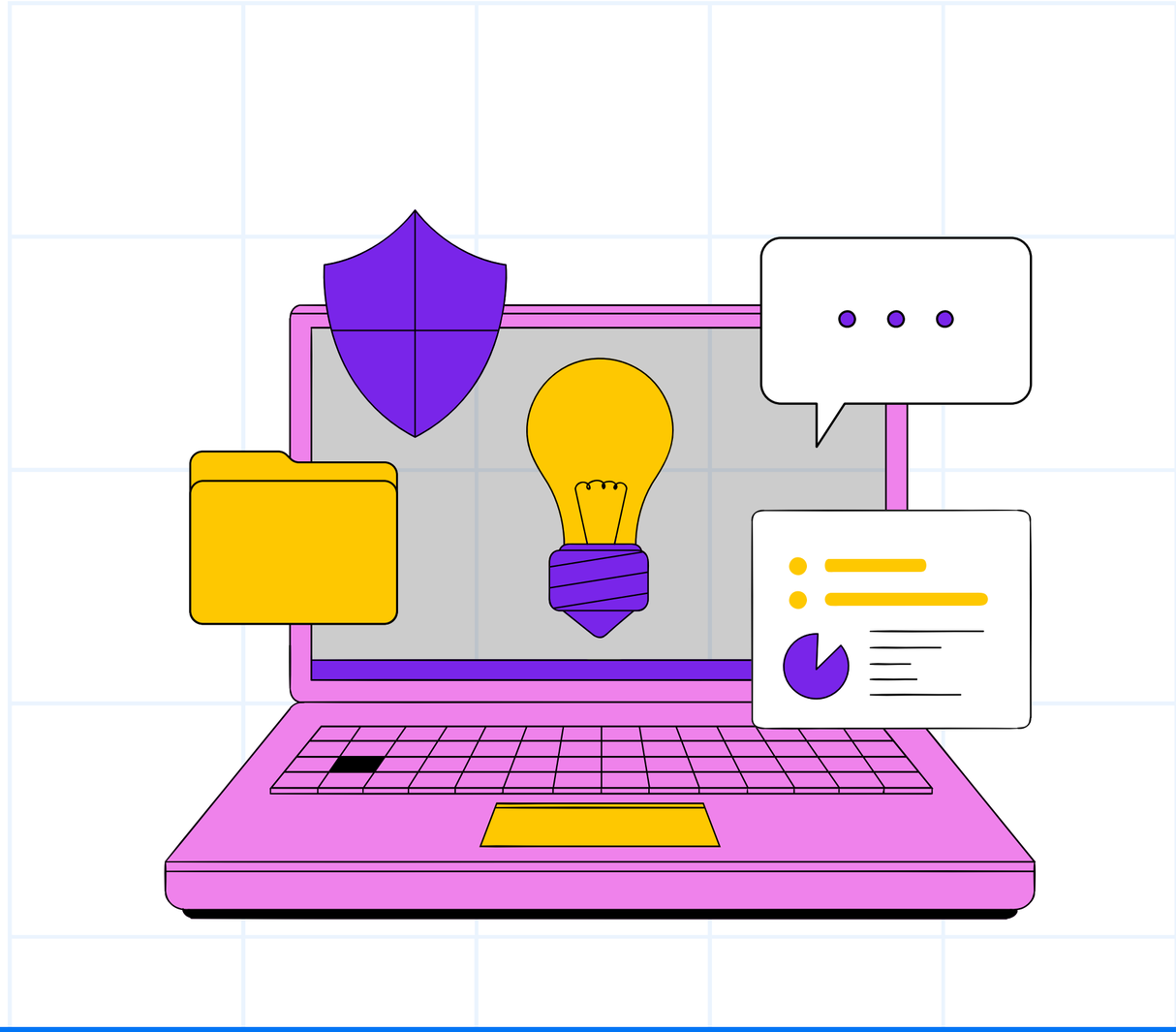
This is why Modular Blockchain is often considered the future of blockchain technology.

The present status of aelf

As elaborated before, aelf operates on a mainchain-sidechain architecture. Its sidechains resemble the consumer chains of Cosmos or the parachains of Polkadot. Similar to these platforms, aelf decouples functionality at the code level — a modular or deconstructed approach rooted in software engineering principles. The primary objectives of this architecture are to facilitate the development of modular plugins within the aelf ecosystem and to enable the loading of various modules in a pluggable manner, thereby enhancing aelf's scalability and flexibility. For instance, the consensus functionality can be treated as a pluggable/replaceable code package, making it convenient for developers to customise.

Notwithstanding the aforementioned points, the current mainchain-sidechain architecture of aelf experiences performance bottlenecks, particularly in terms of its TPS capacity. aelf will be introducing a new Layer 2 chain utilising ZK Rollup technology that would considerably enhance aelf's performance, scalability, and compatibility with the Ethereum Virtual Machine (EVM).

Consequently, we have initiated the **Modular + ZK Rollup** project to address these challenges.



The aelf solution

We are building a ZK-rollup on aelf, based on the Polygon CDK, operating in the validium mode. Its primary components include:

Executor/Verifier

zkEVM / ZKProver

Data Availability

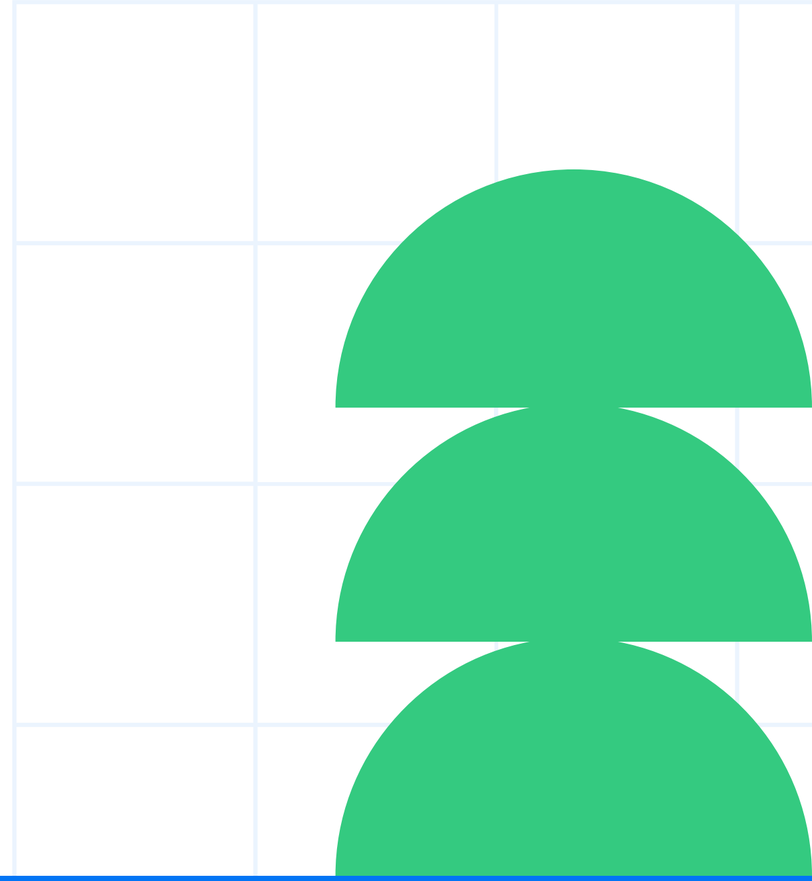
External DA + Data Availability Committee (DAC)

Consensus

On-chain Smart Contracts

Sequencer

Bridge



Verification Process

The aelf ZK-rollup utilises Polygon CDK, an advanced open-source framework designed for rapidly deploying ZK-powered Layer 2 (L2) blockchains on Ethereum, with significant adaptation and optimisation efforts.

Operating in the validium mode while maintaining seamless interoperability with all other EVM chains, aelf ZK-rollup offers a high-performance Layer 2 scaling solution. As transaction data is not stored on the aelf mainnet but executed and stored off-chain, this significantly enhances scalability. Validium, under the premise of ensuring security, reduces L1 gas storage costs, thereby lowering transaction costs for users on Layer 2 while significantly improving user privacy and experience.

Unlike traditional rollups and sidechains, aelf ZK-rollup only shares proofs of validity, which are used to confirm transaction outcomes with Layer 1, rather than the actual executed transaction data. Its operation works as follows: A verifier smart contract is deployed on aelf Layer 1, and validium submits proofs of validity to this contract. These proofs, inherently zero-knowledge, contain transaction results without revealing specific transaction data.

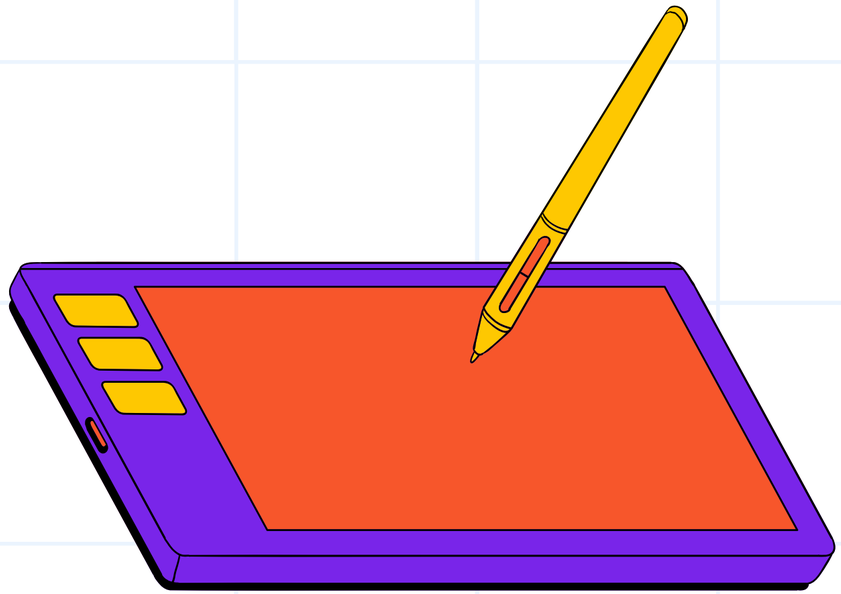
The verifier smart contract assesses the validity of the proofs. Should any batch submitted by validium be found invalid, it will be rejected and not stored on the aelf Layer 1.

Consensus Contract

ZkEVM.cs is the underlying protocol that ensures the correctness of state changes by using validity proofs. To confirm adherence to specific predefined rules for state transitions, the consensus contract (ZkEVM.cs, deployed on aelf Layer 1) comes into play.

ZkEVM.cs is responsible for verifying validity proofs to confirm that each transaction has been executed correctly, using ZK-SNARK circuits for validation. For the system to function properly, two key processes are involved: **transaction batching** and **transaction verification**. To carry out these processes, aelf ZK-rollup involves two types of participants: **sequencers** and **aggregators**.

1. **Sequencer**: Proposes transaction batches to the network, grouping and adding transaction requests to ZkEVM.cs.
2. **Aggregator**: Reviews the validity of transaction batches and provides necessary validity proofs. Permissionless aggregators can submit these proofs to attest to the accuracy of state transition computations.



Data Availability

aelf ZK-rollup's validium mode incorporates a secure data availability layer managed by the DAC, which:

1. Verifies the availability of data associated with specific blockchain blocks; and
2. Ensures data robustness and computational efficiency for aelf ZK-rollup.

Advantages of DAC include:

1. Reduce transaction fees; less computation equals lower costs.
2. State privacy; maintaining secure records of state changes for data integrity.

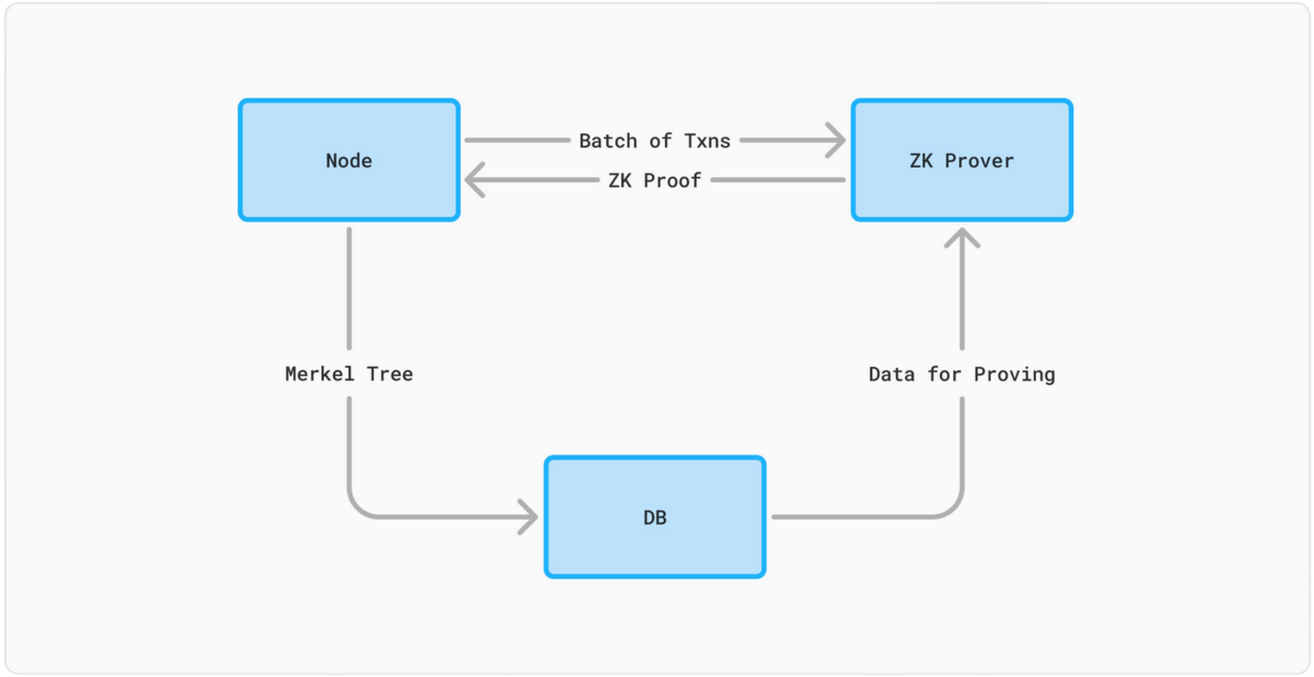
aelf ZK-rollup Node

aelf ZK-rollup Node is used to process transactions, synchronizes states, generates, and submits proofs. Sequencers and aggregators manage L2 state and finality on L1.

zkEVM / ZKProver

One of the core functionalities of aelf ZK-rollup's nodes is its zkEVM and ZKProver. The general operation flow is as follows: zkEVM / ZKProver receives a large batch of transactions from Layer 2, executes the transactions, and utilises ZK (Zero-Knowledge) technology to generate zero-knowledge proofs. The proof generation process is outlined below:

1. The node sends the contents of the Merkle tree to the database for storage.
2. Subsequently, the node sends a batch of transactions to the ZKProver component.
3. ZKProver accesses the database and retrieves necessary information to generate verifiable proofs for the transactions sent by the node. This information includes the Merkle root, keys, and hash values of relevant sibling nodes.
4. Finally, ZKProver generates proofs for the transactions and sends them back to the node.



Sequencer

The Sequencer is responsible for ordering transactions and constantly updating the global state:

1. Transaction Ordering: Retrieves transactions from the pool and adds them to the state.
2. State Transition: Collaborates with the Executor to process transactions and update the state.
3. Trustworthy Finality: Once the Sequencer adds a transaction to the state, it shares this information with other nodes, making the transaction final. Other nodes need to trust that the transaction has been added to the state until they receive data availability (DA) and validity (ZKP) confirmations, as detailed below in Transaction Journey on aelf ZK-Rollup.

Aggregator

The Aggregator is responsible for submitting proofs of the validity of the Layer 2 state to Layer 1. To achieve this, it takes batches ordered by the Sequencer and interacts with Provers to generate Zero-Knowledge Proofs (ZKPs). To efficiently accomplish this, the Aggregator will:

1. Coordinate with one or more Provers
2. Aggregate proofs for multiple batches, where a single proof can validate multiple batches

Bridge

Token asset access and contract invocations (message passing) between aelf's Layer 2 and Layer 1 are facilitated through the zkEVM Bridge smart contract. It achieves network interoperability by utilising two distinct Global Exit Tree Root managers on Layer 1 and Layer 2, along with the bridging smart contract (ZkEVMBridge.cs).

Gas Fees

aelf ZK-rollup involves two primary participants who both earn and pay ELF tokens as gas fee.

Sequencer:

- Pays gas fees for creating and proposing batches.
- Receives fees paid by transaction requesters or network users as rewards when successfully proposing valid batches containing valid transactions.

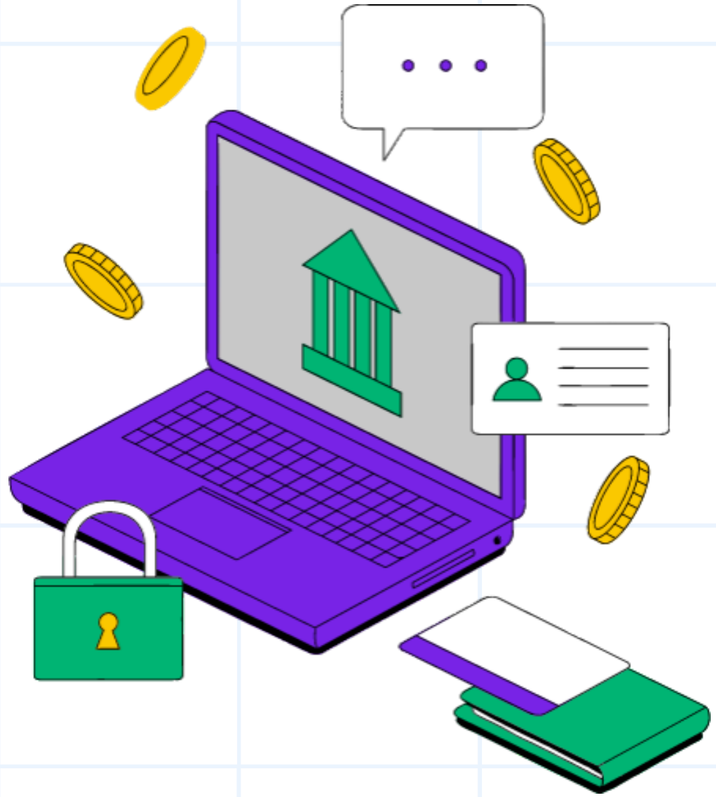
Aggregator:

- Validates transactions proposed and batched by the Sequencer.
- Runs on the aelf ZK-rollup's ZKNode software and utilises ZKProver to create zero-knowledge validity proofs.
- Provides validity proofs to verify transactions, earning ELF fees paid by the Sequencer for each successful verification.

Transaction Journey on aelf ZK-rollup

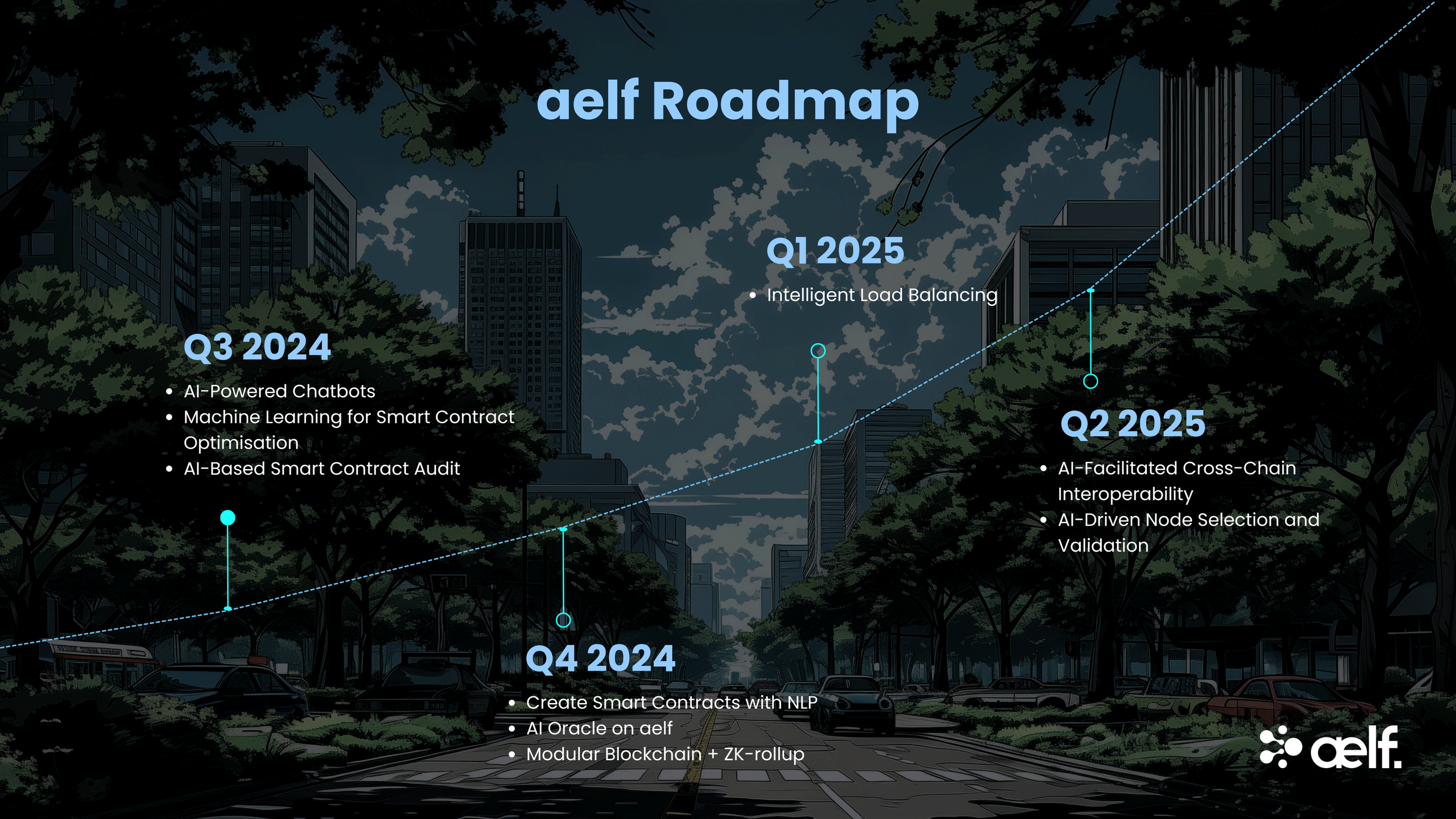
Before conducting Layer 2 transactions, users need to transfer some ELF from L1 to L2 through the aelf ZK-rollup bridge for gas fees. The subsequent steps are as follows:

1. Users initiate a transaction from their wallet (e.g. Portkey) and send it to the Sequencer.
2. Once the Sequencer commits to adding the transaction, it is finalised on L2.
3. At this point, the transaction is settled on L2, but its state has not yet propagated to L1. This state is known as the **Pre-validated State**.
4. The Sequencer transfers batch data to a smart contract on L1, allowing any node to safely and trustlessly synchronise its state from L1. This state is called the **Committed State**.
5. The Aggregator collects pending transactions for validation and constructs proofs to achieve finality on L1.
6. Once the proofs are verified, the user's transaction achieves finality on L1, a crucial step for withdrawals and other operations. This state is referred to as the **Verified State**.



Roadmap

This roadmap outlines aelf's strategic integration of AI technologies, focusing on enhancing user and developer experiences, optimising smart contract operations, and ensuring robust, scalable blockchain solutions that is on par with leading technologies.


aelf Roadmap
Q3 2024
AI-Powered Chatbots
Machine Learning for Smart Contract Optimisation
AI-Based Smart Contract Audit
Q4 2024
Create Smart Contracts with NLP
AI Oracle on aelf
Modular Blockchain + ZK-rollup
Q1 2025
Intelligent Load Balancing
Q2 2025
AI-Facilitated Cross-Chain Interoperability
AI-Driven Node Selection and Validation
aelf.

Q3 2024

- **Enhancing User and Developer Experience with AI-Powered Chatbots on aelf**
  - Integrating AI-powered chatbots into the aelf blockchain ecosystem enhances user and developer experiences by providing real-time support, automating routine tasks, and facilitating easy access to information.

- **Machine Learning Models for Smart Contract Optimisation**
  - Develop and deploy machine learning models to optimise the execution and gas usage of smart contracts, improving overall efficiency and performance on the aelf platform.

- **AI-Based Smart Contract Audit for Developers on aelf**
  - Launch an aelf AI-powered auditing tool to analyse smart contract code, identify vulnerabilities, and offer remediation suggestions before deployment, enhancing security and reliability.

Q4 2024

- **Enabling Non-Technical Users to Create Smart Contracts with NLP on aelf**
  - Introduce Natural Language Processing (NLP) tools to simplify the creation of smart contracts on aelf, making blockchain technology accessible to non-technical users and streamlining the development process.

- **AI Oracle on aelf**
  - Deploy an AI based oracle to facilitate the development of AI-powered dApps, improve interaction capabilities, and support collaborations with AI Agent platforms on the aelf blockchain.

- **aelf as a Modular + ZK-rollup Layer 2 Blockchain**
  - Advance aelf's modular architecture to allow for greater customisation and scalability, catering to diverse application needs and enhancing overall network flexibility, while implementing ZK-Rollup as a Layer 2 solution for optimal TPS, efficiency and EVM compatibility.

Q1 2025

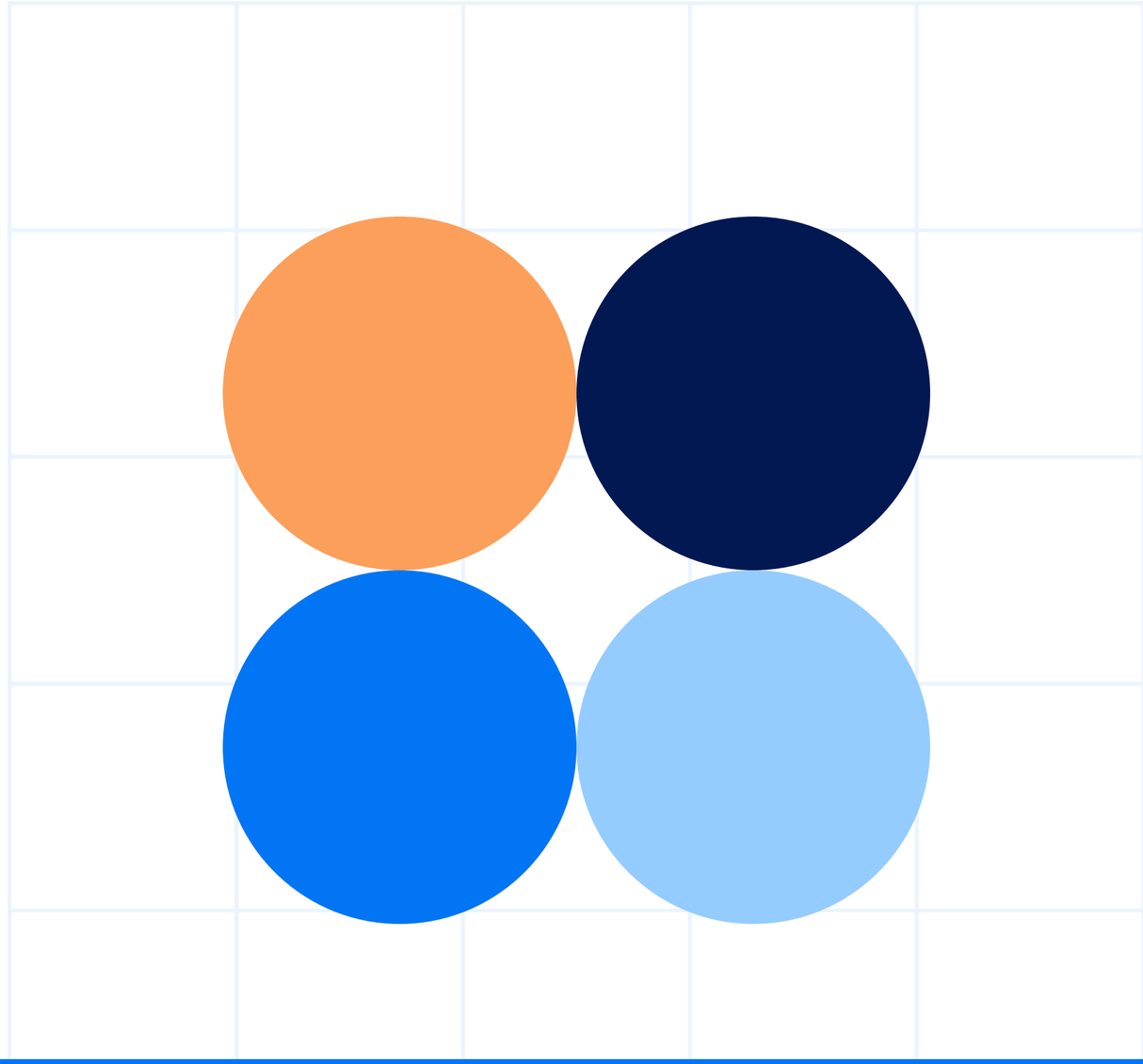
- **Intelligent Load Balancing for aelf**
  - Integrate AI-driven intelligent load balancing on aelf to ensure efficient traffic distribution and resource utilisation across the network, enhancing performance and reliability.

Q2 2025

- **AI-Facilitated Cross-Chain Interoperability for aelf**
  - Establish AI-driven mechanisms on aelf to facilitate seamless cross-chain interoperability, expanding aelf's connectivity with other blockchain networks and enhancing overall functionality.
- **AI-Driven Node Selection and Validation for aelf**
  - (TBD) Implement AI algorithms for node selection and validation, improving network security, efficiency, and reliability.

References and further readings

1. aelf. Whitepaper V 1.7.0. October 25th, 2022. [Link]
2. Blockchain Council. "How Blockchain and AI Can Work Together." Blockchain Council, January 22, 2024. Retrieved from [Link]
3. Buterin, V. (2024, January 30). Crypto AI. Retrieved from [Link]
4. Chen, Y., & Bellavitis, C. (2020). Blockchain disruption and decentralised finance: The rise of decentralised business models. Journal of Business Venturing Insights, 13, e00151. [Link]
5. Christos Smiliotopoulos, Ryan Benton, Georgios Kambourakis, Dimitrios Damopoulos. "Blockchain and Artificial Intelligence: Synergies and Conflicts." *arXiv*, 2024. [Link]
6. Christos Smiliotopoulos, Ryan Benton, Georgios Kambourakis, Dimitrios Damopoulos. "The Convergence of Artificial Intelligence and Blockchain: The State of Play and the Road Ahead." *MDPI Information*, vol. 15, no. 5, 2024. [Link]
7. Cointelegraph. "EU Commission Urged to Prepare for Blockchain and AI Integration." Cointelegraph, May 24, 2024. Retrieved from [Link]
8. Dinh, T. N., & Thai, M. T. (2018). AI and blockchain: A disruptive integration. Computer, 51(9), 48-53. [Link]
9. ETH Research. (2024). How AI Revolutionizes Ethereum: Another View on AI & Blockchain. Retrieved from [Link]
10. Pandl, K. D., Thiebes, S., Schmidt-Kraepelin, M., & Sunyaev, A. (2020). On the convergence of artificial intelligence and distributed ledger technology: A scoping review and future research agenda. IEEE Access, 8, 57075-57095. [Link]
11. Polygon Technology. (2024). Polygon CDK Documentation. Retrieved from [Link]
12. Salah, K., Rehman, M. H. U., Nizamuddin, N., & Al-Fuqaha, A. (2019). Blockchain for AI: Review and open research challenges. IEEE Access, 7, 10127-10149. [Link]
13. Techopedia. (2024). Why Is the Future of Blockchain Modular? Retrieved from [Link]
14. World Economic Forum. "How Immersive Technology, Blockchain and AI Are Converging." World Economic Forum, June 2024. Retrieved from [Link]
15. Xiong, Z., Zhang, Y., Niyato, D., Wang, P., & Han, Z. (2018). When mobile blockchain meets edge computing. IEEE Communications Magazine, 56(8), 33-39. [Link]
16. Zheng, Z., Xie, S., Dai, H. N., Chen, W., Chen, X., Weng, J., & Imran, M. (2020). An overview on smart contracts: Challenges, advances and platforms. Future Generation Computer Systems, 105, 475-491. [Link]



Experience AI + Blockchain of Tomorrow

Learn more about aelf

www.aelf.com

Contact us to get started today

form.aelf.com/contact