

Exploring Ethereum: Clients and the EVM

Ethereum was created through the joint efforts of Vitalik Buterin and several other co-founders—Gavin Wood, Charles Hoskinson, Jeffrey Wilcke, Mihai Alisie, Amir Chetrit, and Anthony Di Iorio. Although Buterin is best known as the main mind behind the idea and the author of the Ethereum whitepaper, the platform's growth and achievements came from the teamwork of the entire founding group.

Ethereum:

- ❖ Ethereum is a blockchain that functions like a global computer.
- ❖ It serves as the base layer for creating decentralized applications and organizations without the need for permission or fear of censorship.
- ❖ At its core is the Ethereum Virtual Machine (EVM) — a single, shared computer whose state is agreed upon by all participants in the network.
- ❖ Every Ethereum node maintains its own copy of the EVM's current state.
- ❖ Anyone on the network can send a request for the EVM to run any kind of computation.
- ❖ When such a request is made, other network participants verify, validate, and execute it.
- ❖ Executing the request changes the EVM's state, and this updated state is recorded and shared across the network.
- ❖ These computation requests are known as transactions, and the full record of all transactions — along with the EVM's current state — is stored on the blockchain, which is maintained and agreed upon by all nodes.

Ethereum Client

- ❖ An Ethereum client is a software application that
- ❖ Follows the rules of the Ethereum protocol.
- ❖ Communicates with other nodes across the network
- ❖ Checks and shares transactions with the network.
- ❖ Keeps an up-to-date copy of the blockchain's state.
- ❖ Lets developers and users interact with the blockchain through RPC APIs.

Types of Ethereum Clients:

- * Ethereum has multiple client implementations — all follow the same protocol but are written in different languages.
- * Execution Clients (EVM-focused)
- * Geth (Go Ethereum) — written in Go.
- * Nethermind - written in C#.
- * Besu - Java-based.
- * Erigon - Go-based, optimized for archival nodes.
- * They execute transactions, maintain state, and run the EVM.

Consensus Clients (Proof-of-Stake era)

- * Since The Merge (2022), Ethereum uses two layers:
- * Execution layer (runs the EVM, processes transactions).
- * Consensus layer (validates blocks, runs PoS).
- * Consensus clients:
- * Prysm — Go.
- * Lighthouse — Rust.
- * Teku — Java.
- * Nimbus — Nim.
- * These talk to execution clients via Engine API (secured with JWT tokens).

EVM – Ethereum Virtual Machine

The EVM is:

- A runtime environment that executes smart contract code.
- Deterministic: given the same inputs, all nodes get the same output.
- Stack-based architecture.
- Works with EVM bytecode compiled from Solidity, Vyper, etc.
- Runs in an isolated sandbox (no direct access to OS, file system, or internet).

Main Features of the EVM

- + **Gas system:** Every operation requires a gas fee, which prevents endless loops and network spam.
- + **Deterministic processing:** All network nodes execute the same instructions to ensure they stay in agreement.
- + **State handling:**
 - **Account state:** Holds balances, transaction counts (nonces), smart contract code, and storage.
 - **World state:** A complete record mapping every address to its account information.

How the Client and EVM Operate Together

Transaction process:

1. A user or DApp submits a transaction, which is picked up by the execution client.
2. The execution client:
 - Checks the transaction's validity (signature, nonce, gas).
 - Processes it inside the EVM.
 - Updates the blockchain state (balances, storage, and smart contract code).
3. The execution client forwards the block details to the consensus client.
4. The consensus client:
 - Performs Proof-of-Stake checks.

Ether (ETH):

- Ether is Ethereum's built-in cryptocurrency.
- Its main role is to create a marketplace for computation, giving participants a financial reason to verify transactions, run smart contracts, and contribute computing power to the network.
- Anyone sending a transaction must include a payment in ETH as an incentive.
- A portion of this payment is burned, while the rest is given to the validator who processes, verifies, and records the transaction on the blockchain.
- The ETH cost depends on how much computation the transaction requires.
- ETH also secures the network in three key ways:
 1. **Validator rewards:** Validators earn ETH for proposing blocks and exposing dishonest activity.
 2. **Staking collateral:** Validators lock up ETH as a guarantee of honest behavior — if they act maliciously, their stake can be slashed (destroyed).
 3. **Consensus voting power:** Staked ETH determines the weight of a validator's vote in selecting the correct chain during consensus.