

Introduction to Ethereum governance {#introduction}

If no one owns Ethereum, how are decisions about past and future changes to Ethereum made? Ethereum governance refers to the process that allows such decisions to be made.

What is governance? {#what-is-governance}

Governance is the systems in place that allow decisions to be made. In a typical organizational structure, the executive team or a board of directors may have the final say in decision-making. Or perhaps shareholders vote on proposals to enact change. In a political system, elected officials may enact legislation that attempts to represent their constituents' desires.

Decentralized governance {#decentralized-governance}

No one person owns or controls the Ethereum protocol, but decisions still need to be made about implementing changes to best ensure the longevity and prosperity of the network. This lack of ownership makes traditional organizational governance an incompatible solution.

Ethereum Governance {#ethereum-governance}

Ethereum governance is the process by which protocol changes are made. It's important to point out that this process isn't related to how people and applications use the protocol - Ethereum is permissionless. Anyone from anywhere in the world can participate in on-chain activities. There are no rules set for who can or cannot build an application or send a transaction. However, there is a process to propose changes to the core protocol, which decentralized applications run on top of. Since so many people depend on Ethereum's stability, there is a very high coordination threshold for core changes, including social and technical processes, to ensure any changes to Ethereum are secure and widely supported by the community.

On-chain vs off-chain governance {#on-chain-vs-off-chain}

Blockchain technology allows for new governance capabilities, known as on-chain governance. On-chain governance is when proposed protocol changes are decided by a stakeholder vote, usually by holders of a governance token, and voting happens on the blockchain. With some forms of on-chain governance, the proposed protocol changes are already written in code and implemented automatically if the stakeholders approve the changes via signing a transaction.

The opposite approach, off-chain governance, is where any protocol change decisions happen through an informal process of social discussion, which, if approved, would be implemented in code.

Ethereum governance happens off-chain with a wide variety of stakeholders involved in the process.

Whilst at the protocol level Ethereum governance is off-chain, many use cases built on top of Ethereum, such as DAOs, use on-chain governance.

More on DAOs

Who is involved? {#who-is-involved}

There are various stakeholders in the [Ethereum community](#), each playing a role in the governance process. Starting from the stakeholders furthest from the protocol and zooming in, we have:

- **Ether holders:** these people hold an arbitrary amount of ETH.[More on ETH.](#)
- **Application Users:** these people interact with applications on the Ethereum blockchain.
- **Application/Tooling Developers:** these people write applications that run on the Ethereum blockchain (e.g. DeFi,

NFTs, etc.) or build tooling to interact with Ethereum (e.g. wallets, test suites, etc.). [More on dapps](#).

- **Node Operators:** these people run nodes that propagate blocks and transactions, rejecting any invalid transaction or block that they come across. [More on nodes](#).
- **EIP Authors:** these people propose changes to the Ethereum protocol, in the form of Ethereum Improvement Proposals (EIPs). [More on EIPs](#).
- **Validators:** these people run nodes that can add new blocks to the Ethereum blockchain.
- **Protocol Developers** (a.k.a. "Core Developers"): these people maintain the various Ethereum implementations (e.g. go-ethereum, Nethermind, Besu, Erigon at the execution layer or Prysm, Lighthouse, Nimbus, Teku, Lodestar at the consensus layer). [More on Ethereum clients](#).

Note: any individual can be part of multiple of these groups (e.g. a protocol developer could champion an EIP, and run a beacon chain validator, and use DeFi applications). For conceptual clarity, it is easiest to distinguish between them, though.

What is an EIP? {#what-is-an-eip}

One important process used in Ethereum governance is the proposal of **Ethereum Improvement Proposals (EIPs)**. EIPs are standards specifying potential new features or processes for Ethereum. Anyone within the Ethereum community can create an EIP. If you're interested in writing an EIP or participating in peer-review and/or governance, see:

More on EIPs

The formal process {#formal-process}

The formal process for introducing changes to the Ethereum protocol is as follows:

1. **Propose a Core EIP:** as described in [EIP-1](#), the first step to formally proposing a change to Ethereum is to detail it in a Core EIP. This will act as the official specification for an EIP that Protocol Developers will implement if accepted.
2. **Present your EIP to Protocol Developers:** once you have a Core EIP for which you've gathered community input, you should present it to Protocol Developers. You can do so by proposing it for discussion on an [AllCoreDevs call](#). It is likely some discussions will have already happened asynchronously on the [Ethereum Magician's forum](#) or in the [Ethereum R&D Discord](#).

Potential outcomes of this stage are:

- The EIP will be considered for a future network upgrade
 - Technical changes will be requested
 - It may be rejected if it is not a priority or the improvement is not large enough relative to the development effort
1. **Iterate towards a final proposal:** after receiving feedback from all relevant stakeholders, you will likely need to make changes to your initial proposal to improve its security or better meet the needs of various users. Once your EIP has incorporated all the changes you believe are necessary, you will need to present it again to Protocol Developers. You will then move to the next step of this process, or new concerns will emerge, requiring another round of iterations on your proposal.
 2. **EIP Included in Network Upgrade:** assuming the EIP is approved, tested and implemented, it gets scheduled as part of a network upgrade. Given the high coordination costs of network upgrades (everyone needs to upgrade simultaneously), EIPs are generally bundled together in upgrades.
 3. **Network Upgrade Activated:** after the network upgrade is activated, the EIP will be live on the Ethereum network.
Note: network upgrades are usually activated on testnets before being activated on the Ethereum Mainnet.

This flow, while very simplified, gives an overview of the significant stages for a protocol change to be activated on Ethereum. Now, let's look at the informal factors at play during this process.

The informal process {#informal-process}

Understanding prior work {#prior-work}

EIP Champions should familiarise themselves with prior work and proposals before creating an EIP which can be seriously considered for deployment on the Ethereum Mainnet. This way, the EIP hopefully brings something new which hasn't been rejected before. The three main places to research this are the [EIP repository](#), [Ethereum Magicians](#) and [ethresear.ch](#).

Working groups {#working-groups}

The initial draft of an EIP is unlikely to be implemented on the Ethereum Mainnet without edits or changes. Generally, EIP Champions will work with a subset of Protocol Developers to specify, implement, test, iterate, and finalize their proposal. Historically, these working groups have required several months (and sometimes years!) of work. Similarly, EIP Champions for such changes should involve relevant Application/Tooling Developers early in their efforts to gather end-user feedback and mitigate any deployment risks.

Community consensus {#community-consensus}

While some EIPs are straightforward technical improvements with minimal nuance, some are more complex and come with tradeoffs which will affect different stakeholders in different ways. This means some EIPs are more contentious within the community than others.

There is no clear playbook on how to handle contentious proposals. This is a result of Ethereum's decentralized design whereby no single stakeholder group can coerce the other through brute force: protocol developers can choose not to implement code changes; node operators can choose not to run the latest Ethereum client; application teams and users can choose not to transact on the chain. Since Protocol Developers have no way to force people to adopt network upgrades, they will generally avoid implementing EIPs where the contentiousness outweighs the benefits to the broader community.

EIP Champions are expected to solicit feedback from all relevant stakeholders. If you find yourself the champion of a contentious EIP, you should try and address objections to build consensus around your EIP. Given the size and diversity of the Ethereum community, there isn't a single metric (e.g. a coin vote) that can be used to gauge community consensus, and EIP Champions are expected to adapt to the circumstances of their proposal.

Beyond the security of the Ethereum network, significant weight has historically been placed by Protocol Developers on what Application/Tooling Developers and Application Users value, given that their using and developing on Ethereum is what makes the ecosystem attractive for other stakeholders. Additionally, EIPs need to be implemented across all client implementations, which are managed by distinct teams. Part of this process usually means convincing multiple teams of Protocol Developers that a particular change is valuable and that it helps end-users or solves a security issue.

Handling disagreements {#disagreements}

Having many stakeholders with different motivations and beliefs means that disagreements are not uncommon.

Generally, disagreements are handled with long-form discussion in public forums to understand the root of the problem and allow anyone to weigh in. Typically, one group concedes, or a happy medium is achieved. If one group feels strongly enough, forcing through a particular change could result in a chain split. A chain split is when some stakeholders protest implementing a protocol change resulting in different, incompatible versions of the protocol operating, from which two distinct blockchains emerge.

The DAO fork {#dao-fork}

Forks are when major technical upgrades or changes need to be made to the network and change the "rules" of the protocol. [Ethereum clients](#) must update their software to implement the new fork rules.

The DAO fork was in response to the [2016 DAO attack](#) where an insecure [DAO](#) contract was drained of over 3.6 million ETH in a hack. The fork moved the funds from the faulty contract to a new contract allowing anyone who lost funds in the hack to recover them.

This course of action was voted on by the Ethereum community. Any ETH holder was able to vote via a transaction on [a](#)

[voting platform](#). The decision to fork reached over 85% of the votes.

It's important to note that whilst the protocol did fork to revert the hack, the weight the vote carried in deciding to fork is debatable for a few reasons:

- The turnout to vote was incredibly low
- Most people didn't know the vote was happening
- The vote only represented ETH holders, not any of the other participants in the system

A subset of the community refused to fork, largely because they felt the DAO incident wasn't a defect in the protocol. They went on to form [Ethereum Classic](#).

Today, the Ethereum community has adopted a policy of non-intervention in cases of contract bugs or lost funds to maintain the credible neutrality of the system.

Watch more on the DAO hack:

The utility of forking {#forking-utility}

The Ethereum/Ethereum Classic fork is an excellent example of a healthy fork. We had two groups who disagreed strongly enough with each other on some core values to feel it was worth the risks involved to pursue their specific courses of action.

The ability to fork in the face of significant political, philosophical or economic differences plays a large part in the success of Ethereum governance. Without the ability to fork the alternative was ongoing in-fighting, forced reluctant participation for those who eventually formed Ethereum Classic and an increasingly differing vision of how success for Ethereum looks.

Beacon Chain governance {#beacon-chain}

The Ethereum governance process often trades off speed and efficiency for openness and inclusivity. In order to accelerate the development of the Beacon Chain, it was launched separately from the proof-of-work Ethereum network and followed its own governance practices.

While the specification and development implementations have always been fully open source, the formal processes used to propose updates described above weren't used. This allowed changes to be specified and agreed upon quicker by researchers and implementers.

When the Beacon Chain merged with the Ethereum execution layer on September 15th, 2022 The Merge was complete as part of the [Paris network upgrade](#). The proposal [EIP-3675](#) was changed from 'Last Call' to 'Final', completing the transition to proof-of-stake.

More on The Merge

How can I get involved? {#get-involved}

- [Propose an EIP](#)
- [Discuss current proposals](#)
- [Get involved in R&D discussion](#)
- [Join the Ethereum R&D discord](#)
- [Run a node](#)
- [Contribute to client development](#)
- [Core Developer Apprenticeship Program](#)

Further reading {#further-reading}

Governance in Ethereum isn't rigidly defined. Various community participants have diverse perspectives on it. Here are a few of them:

- [Notes on Blockchain Governance](#) - Vitalik Buterin

- [How does Ethereum Governance work?](#) – *Cryptotesters*
- [How Ethereum governance works](#) – *Micah Zoltu*
- [What is an Ethereum core developer?](#) - *Hudson Jameson*
- [Governance, Part 2: Plutocracy Is Still Bad](#) - *Vitalik Buterin*
- [Moving beyond coin voting governance](#) - *Vitalik Buterin*