How IPC compares

IPC is a standout framework that strikes a considerable balance, to achieve breakthroughs in scaling.

Highly customizable without compromising security

Most L2 scaling solutions today either inherit the L1's security features but don't have their own consensus algorithms (e.g. rollups), or do the reverse (e.g. sidechains). They are also deployed in isolation and require custom bridges or protocols to transfer assets and state between L2s that share a common L1, which are vulnerable to attacks. In contrast, IPC subnets have their own consensus algorithms, inherit security features from the parent subnet and have native cross-net communication, eliminating the need for bridges.

Multi-chain interoperability

- IPC uses [Tendermint Core](#)
- as a generic blockchain SMR system, without defaulting to the Cosmos SDK (written in Go). This allows IPC to plug in our own application logic regardless of what language it's written in: it can be Go, Rust, Java, Haskell, Scheme, etc.
- IPC uses the [Filecoin Virtual Machine (FVM)](#)
- as its transaction execution layer. The FVM is a WASM-based polyglot execution environment for IPLD data and is designed to support smart contracts written in any programming language, compiled to WebAssembly. This enables multi-chain support and gives developers the flexibility to build with familiar tools. Today, IPC is fully compatible with Filecoin and Ethereum and can use either as a rootnet, with more multi-chain support in the roadmap.
-

Compute-Storage Interoperability with Filecoin and more

IPC is designed to seamlessly integrate with Filecoin and EVM-compatible chains (with more to come), allowing developers to embed IPC subnets within these ecosystems. In particular, IPC unlocks new compute possibilities with the data-centric L1, [Filecoin](#) , which is the largest decentralized storage network. IPC can leverage its storage primitives, like [IPLD](#) data integration, to deliver enhanced solutions for data availability and more.

Increased performance

IPC's modular runtime enables the creation of truly flexible blockchains to increase thoughput while managing gas efficiency. Developers can dynamically adjust their throughput by spawning and closing temporary subnets as needed.

Last updated18 days ago On this page * [Highly customizable without compromising security](#) * [Multi-chain interoperability](#) * [Compute-Storage Interoperability with Filecoin and more](#) * [Increased performance](#)

Was this helpful? [Edit on GitHub](#)