Resources

In this section, we'll provide a comprehensive overview of the key tools and technologies used in the Sei blockchain ecosystem. Whether you are new to web3 development or an experienced developer looking to expand your skills, these resources will help you get started and deepen your understanding. Developers on Sei have the unique advantage of choosing between CosmWasm and EVM for their smart contract development, with the ability to seamlessly interact between the two environments.

Cosmos SDK

The Cosmos SDK is a modular framework for building custom blockchains within the Cosmos ecosystem. It provides a set of tools and libraries that simplify the creation and management of interoperable blockchains.

- Tendermint
- : Tendermint is the consensus engine that powers the Cosmos SDK. It ensures fast and secure consensus through a Byzantine Fault Tolerant (BFT) protocol.* Tendermint Documentation(opens in a new tab)
- Module Structure
- : The Cosmos SDK uses a modular architecture, allowing developers to create and integrate various modules to build feature-rich blockchains.* Module Structure Documentation(opens in a new tab)
- Transaction Structure
- : Understanding the structure of transactions is crucial for developing applications on the Cosmos SDK. Transactions are the primary means of interacting with the blockchain.* Transaction Structure Documentation(opens in a new tab)
- IBC (Inter-Blockchain Communication)
- : IBC is a protocol that enables communication and asset transfers between different blockchains within the Cosmos ecosystem.* IBC Overview(opens in a new tab)
- Event structure
- : ADD THIS

EVM (Ethereum Virtual Machine)

The EVM is the runtime environment for smart contracts on Ethereum and EVM-compatible blockchains like Sei. It allows developers to write and deploy smart contracts using Solidity.

- Solidity
- : Solidity is the most widely used programming language for writing smart contracts on the EVM. It is a statically-typed language that is influenced by JavaScript, Python, and C++.* Solidity Documentation(opens in a new tab)
- Hardhat
- : Hardhat is a development environment for compiling, deploying, testing, and debugging Ethereum software. It is highly extensible and integrates well with other tools.* Hardhat Documentation(opens in a new tab)
- Foundry
- : Foundry is a toolkit for Ethereum application development, providing a comprehensive suite of tools for testing and deploying smart contracts.* Foundry Documentation(opens in a new tab)

CosmWasm

CosmWasm is a smart contract platform built for the Cosmos ecosystem, enabling developers to write smart contracts in WebAssembly (Wasm). Developers on Sei can choose CosmWasm for its safety and performance benefits, and seamlessly interact with EVM smart contracts.

- Rust
- : Rust is the primary programming language used for writing CosmWasm smart contracts. It is known for its safety, performance, and concurrency.* Rust Documentation(opens in a new tab)

@sei-js

- Link to typedocs
- Explain what it is

Last updated onMay 23, 2024 Tools and Resources Differences from Ethereum