Getting started developing for OP Stack chains

This guide explains the basics of OP Stack development. OP Stack chains are EVM equivalent(opens in a new tab) , meaning they run a slightly modified version of the samegeth you run on mainnet. Therefore, the differences between OP Stack development and Ethereum development are minor. But a few differencesdo exist .

OP Stack chains endpoint URLs

To access any Ethereum type network you need an endpoint.These providers support our networks.

Network choice

For development purposes we recommend you use either a local development node orOP Sepolia(opens in a new tab) . That way you don't need to spend real money. If you need ETH on OP Sepolia for testing purposes,you can use this faucet(opens in a new tab) .

Interacting with contracts on OP Stack chains

We have Hardhat's Greeter contract on OP Sepolia at address0x9d334aFBa83865E67a9219830ADA57aaA9406681(opens in a new tab) . You can verify your development stack configuration by interacting with it.

Development stacks

As you can see in the different development stacks below, the way you deploy contracts and interact with them on OP Stack chains is almost identical to the way you do it with L1 Ethereum. The most visible difference is that you have to specify a different endpoint (of course). The list of other differences ishere .

- Apeworx(opens in a new tab)
- Brownie(opens in a new tab)
- Foundry(opens in a new tab)
- Hardhat(opens in a new tab)
- Remix(opens in a new tab)
- Truffle(opens in a new tab)
- Waffle(opens in a new tab)

Best practices

Use provided EVM

It is best to start development with the EVM provided by the development stack. Not only is it faster, but such EVMs often have extra features, such as theability to log messages from Solidity(opens in a new tab)or agraphical user interface(opens in a new tab) .

Debug before deploying

After you are done with that development, debug your decentralized application using either adevelopment node or theSepolia test network . This lets you debug parts that are OP Stack chains specific such as calls to bridges to transfer ETH or tokens between layers.

Only when you have a version that works well on a test network should you deploy to the production network, where every transaction has a cost.

Contract source verification

You don't have to upload your source code toblock explorers , but it is a good idea. On the test network, it lets you issue queries and transactions from the explorer's user interface. On the production network, it lets users know exactly what your contract does, which is conducive to trust.

Just remember, if you usethe Etherscan API(opens in a new tab) , you need one API key for OP Stack chains and a separate one for OP Sepolia.