Chain Operator Overview

This document provides an overview of how to operate an OP Stack chain, including rollup and chain deployment basics. If you want to skip ahead to deploying your chain, you can start with thechain operator tutorials .

Rollup Basics

The OP Stack gives you the ability to deploy your own Rollup chains that use a Layer 1 blockchain to host and order transaction data. So, arollup is the name given to a blockchain that piggy-backs off of the security of another "parent" blockchain. Specifically, OP Stack Rollups take advantage of the consensus mechanism (like PoW or PoS) of their parent chain instead of providing their own. For OP Stack chains, this parent blockchain is Ethereum. For more information on the rollup protocol, seeRollup Protocol Guide .

Deployment Basics

OP Stack chains require four different components for deployment. It's useful to understand each component before you deploy your chain.

1. Smart Contracts
2. : OP Stack chains use several smart contracts on the L1 blockchain to manage aspects of the Rollup. Each OP Stack chain has its own set of L1 smart contracts that are deployed when the chain is created. For more information on smart contracts, seeSmart Contract Overview
3. .
4. Sequencer Nodes
5. : OP Stack chains use Sequencer nodes to gather proposed transactions from users and publish them to the L1 blockchain. OP Stack nodes, like Ethereum nodes, have a consensus client to determine the list and ordering of blocks and of transactions that are part of your blockchain and execution client, in order to execute new submitted transactions and to update the state of the blockchain to reflect these new transactions.
6. Batcher
7. : The Batcher is a service that publishes transactions from the Sequencer to the L1 blockchain. The Batcher runs continuously alongside the Sequencer and publishes transactions in batches (hence the name) on a regular basis.
8. Proposer
9. : The Proposer is a service responsible for publishing transactionsresults
10. (in the form of L2 state roots) to the L1 blockchain. This allows smart contracts on L1 to read the state of the L2, which is necessary for cross-chain communication and reconciliation between state changes.

Client Implementations

- Consensus clients:op-node
- (maintained by Optimism Foundation),magi (opens in a new tab)
- (maintained by a16z) andhildr (opens in a new tab)
- (maintained by OptimismJ)
- Execution clients:op-geth
- (maintained by Optimism Foundation),op-erigon (opens in a new tab)
- (maintained by Test in Prod), andop-nethermind
- (coming soon)

Chain Operator Tutorials

Here's a curated collection of chain operator tutorials put together by the Optimism community. They'll help you get a head start deploying your first OP Stack chain.

Tutorial Name Description Difficulty Level Creating Your Own L2 Rollup Learn how to spin up your own OP Stack testnet chain Medium Using the OP Stack Client SDK Learn how to use the OP Stack Client SDK when working with native and non-native supported chains. Easy Adding Attributes to the Derivation Function Learn how to modify the derivation function for an OP Stack chain to track the amount of ETH being burned on L1. Easy Adding a Precompile Learn how to run an EVM with a new precompile for OP Stack chain operations to speed up calculations that are not currently supported. Easy Modifying Predeployed Contracts Learn how to modify predeployed contracts for an OP Stack chain by upgrading the proxy. Easy Pause and Unpause the Bridge Learn how to pauseOptimismPortal as a backup safety mechanism on your OP Stack chain. Easy You can alsosuggest a new tutorial(opens in a new tab) if you have something specific in mind. We'd love to grow this list!

Next Steps

- After deploying your chain, check therollup operations guide

- for common operations you'll need to run with your rollup.
- If you run into any problems, please visit the Chain Troubleshooting Guide
- for help.