How to Run an Alt-DA Mode Chain

⚠ The Alt-DA Mode feature is a Beta feature of the MIT licensed OP Stack. While it has received initial review from core contributors, it is still undergoing testing and may have bugs or other issues. This guide provides a walkthrough for chain operators who want to run an Alt-DA Mode chain. See theAlt-DA Mode Explainer for a general overview of this OP Stack configuration. An Alt-DA Mode OP Stack chain enables a chain operator to post and read data to any alternative data availability layer that has built a functioning OP Stack DA Server.

Prerequisite

You should use at least the following compatible op* versions when running your chain.

- op-node/v1.8.0-rc.1
- op-proposer/v1.8.0-rc.1
- op-batcher/v1.8.0-rc.1
- Latest version of op-geth(opens in a new tab)

For deploying your contracts, use the latest release.

If you are trying to launch an Alt-DA Mode Chain using a custom gas token, follow the contract deployment instructions in ourCustom Gas Token docs .

Setup Your DA Server

⚠ DA Servers are not built or maintained by Optimism Collective Core Contributors. They are often maintained by the team that built the DA Layer that the DA Server enables access to. Please reach out to the team who built the DA Server you are trying to run with any questions or issues. * Celestia's docs on how to run theCelestia DA server(opens in a new tab)* EigenDA's docs on how to run theEigenDA DA server(opens in a new tab)* Avail's docs on how to run theAvailDA DA Server(opens in a new tab)

Configure Yourop-node

- Spin up your OP chain as usual but set--altda.enabled=true
- and point bothop-batcher
- andop-node
- to the DA server.
- No configuration changes are required forop-geth
- orop-proposer
- .

Alt-DA (EXPERIMENTAL)

--altda.da-server value (OP_NODE_ALTDA_DA_SERVER) HTTP address of a DA Server

--altda.enabled (default: false) (OP_NODE_ALTDA_ENABLED) Enable Alt-DA mode

--altda.verify-on-read (default: true) (OP_NODE_ALTDA_VERIFY_ON_READ) Verify input data matches the commitments from the DA storage service

Configure Your Batcher

- Set--altda.enabled=true
- and--altda.da-service=true
- .
- Provide the URL for--atlda.da-server=DA_SERVER_HTTP_URL
- .

--altda.da-server value (OP_BATCHER_ALTDA_DA_SERVER) HTTP address of a DA Server

--altda.da-service (default: false) (OP_BATCHER_ALTDA_DA_SERVICE) Use DA service type where commitments are generated by the DA server

--altda.enabled (default: false) (OP_BATCHER_ALTDA_ENABLED) Enable Alt-DA mode

--altda.verify-on-read (default: true) (OP_BATCHER_ALTDA_VERIFY_ON_READ) Verify input data matches the commitments from the DA storage service After completing steps 1-3 above, you will have an Alt-DA mode chain up and running.

Set Your Fee Configuration

- Chain operators are not posting everything to Ethereum, just commitments, so chain operators will need to determine fee scalars values to charge users. The fee scalar values are network throughput dependent, so values will need to be adjusted by chain operators as needed.
- Cost structure for Alt-DA Mode: The transaction data is split up into 128kb chunks and then submitted to your DA Layer. Then, 32 byte commitments are submitted (goes to batch inbox address) to L1 for each 128kb chunk. Then, figure out how much that costs relative to the amount of transactions your chain is putting through.
- Set scalar values inside the deploy config. The example below shows some possible fee scalar values, calculated assuming negligible DA Layer costs, but will need to be adjusted up or down based on network throughput - as a reminder of how to set your scalar values, seethis section(opens in a new tab)
- of the docs.
- // Set in Deploy Config
- "gasPriceOracleBaseFeeScalar": 7663, // Approximate commitment tx base cost
- "gasPriceOracleBlobBaseFeeScalar": 0, // blobs aren't used for submitting the small data commitments

Some initial scalar values must be set early on in the deploy config inStep 2 . And then at a later point, chain operators can update the scalar values with an L1 transaction.

For Node Operators (Full and Archive Nodes)

- Run a DA server as laid out inStep 1
- Provide the same--altda.enabled=true, --altda.da-server...
- onop-node
- as listed inStep 2

Next Steps

- Additional questions? See the FAQ section in theAlt-DA Mode Explainer
- .
- For more detailed info on Alt-DA Mode, see thespecs(opens in a new tab)
- .
- If you experience any problems, please reach out todeveloper support(opens in a new tab)
- .

Preinstalls Run a Custom Gas Token Chain