Rollup Configuration

New OP Stack blockchains are currently configured with a JSON file inside the Optimism repository. You can see example configurations in thedeploy-config directory(opens in a new tab).

⚠ The Rollup configuration is an active work in progress and will likely evolve significantly as time goes on. If something isn't working about your configuration, you can refer to thesource code(opens in a new tab). Standard configuration is the set of requirements for an OP Stack chain to be considered a Standard Chain within the superchain. These requirements are currently a draft, pending governance approval. For more details, please see thisgovernance thread(opens in a new tab).

Configuration Values

Offset Values

These offset values determine when network upgrades (hardforks) activate on your blockchain.

l2GenesisRegolithTimeOffset

L2GenesisRegolithTimeOffset is the number of seconds after genesis block that Regolith hard fork activates. Set it to 0 to activate at genesis. Nil to disable Regolith.

- Type:
- Number of seconds
- Default:
- nil

l2GenesisCanyonTimeOffset

L2GenesisCanyonTimeOffset is the number of seconds after genesis block that Canyon hard fork activates. Set it to 0 to activate at genesis. Nil to disable Canyon.

- Type:
- Number of seconds
- Default:
- nil

l2GenesisDeltaTimeOffset

L2GenesisDeltaTimeOffset is the number of seconds after genesis block that Delta hard fork activates. Set it to 0 to activate at genesis. Nil to disable Delta.

- Type:
- Number of seconds
- Default:
- nil

l2GenesisEcotoneTimeOffset

L2GenesisEcotoneTimeOffset is the number of seconds after genesis block that Ecotone hard fork activates. Set it to 0 to activate at genesis. Nil to disable Ecotone.

- Type:
- Number of seconds
- Default:
- nil

l2GenesisFjordTimeOffset

L2GenesisFjordTimeOffset is the number of seconds after genesis block that Fjord hard fork activates. Set it to 0 to activate at genesis. Nil to disable Fjord.

- Type:
- Number of seconds
- Default:
- nil

l2GenesisInteropTimeOffset

L2GenesisInteropTimeOffset is the number of seconds after genesis block that the Interop hard fork activates. Set it to 0 to activate at genesis. Nil to disable Interop.

- Type:
- Number of seconds
- Default:
- nil

l1CancunTimeOffset

When Cancun activates. Relative to L1 genesis.

- Type:
- Number of seconds
- Default:
- None

Admin addresses

finalSystemOwner

FinalSystemOwner is the L1 system owner. It owns any ownable L1 contracts.

- Type:
- L1 Address
- Default value:
- None
- Recommended value:
- It is recommended to have a single admin
- address to retain a common security model.
- Notes:
- Must not beaddress(0)

proxyAdminOwner

ProxyAdmin contract owner on the L2, which owns all of the Proxy contracts for every predeployed contract in the range0x42...0000 to0x42...2048 . This makes predeployed contracts easily upgradeable.

- Type:
- L2 Address
- Default value:
- None
- Recommended value:
- It is recommended to have a single admin
- address to retain a common security model.
- Notes:
- Must not beaddress(0)

Proxy Addresses

l1StandardBridgeProxy

L1StandardBridgeProxy represents the address of the L1StandardBridgeProxy on L1 and is used as part of building the L2 genesis state.

- Type:
- L1 Address
- Default value:
- None
- Notes:
- Must not beaddress(0)

l1CrossDomainMessengerProxy

L1CrossDomainMessengerProxy represents the address of the L1CrossDomainMessengerProxy on L1 and is used as part of building the L2 genesis state.

- Type:
- L1 Address
- Default value:
- None
- Notes:
- Must not beaddress(0)

l1ERC721BridgeProxy

L1ERC721BridgeProxy represents the address of the L1ERC721Bridge on L1 and is used as part of building the L2 genesis state.

- Type:
- L1 Address
- Default value:
- None
- Notes:
- Must not beaddress(0)

systemConfigProxy

SystemConfigProxy represents the address of the SystemConfigProxy on L1 and is used as part of the derivation pipeline.

- Type:
- L1 Address
- Default value:
- None
- Notes:
- Must not beaddress(0)

optimismPortalProxy

OptimismPortalProxy represents the address of the OptimismPortalProxy on L1 and is used as part of the derivation pipeline.

- Type:
- L1 Address
- Default value:
- None
- Notes:
- Must not beaddress(0)

Blocks

These fields apply to L2 blocks: Their timing, when they need to be written to L1, and how they get written.

l2BlockTime

Number of seconds between each L2 block. Must be < = L1 block time (12 on mainnet and Sepolia).

- Type:
- Number of seconds
- Default value:
- None
- Notes:
- Must not be0
- . Must be less than the L1 blocktime and a whole number.
- Standard Config Requirement:
- 2 seconds. High security and
- interoperability compatibility requirement, until de-risked/solved at app
- layer.

maxSequencerDrift

How far the L2 timestamp can differ from the actual L1 timestamp.

- Type:
- Number of seconds

- Default value:
- None
- Notes:
- Must not be0
- . 1800 (30 minutes) is the constant that takes
- effect with theFjord activation
- .

sequencerWindowSize

Maximum number of L1 blocks that a Sequencer can wait to incorporate the information in a specific L1 block. For example, if the window is 10 then the information in L1 block n must be incorporated by L1 block n+10.

- Type:
- Number of blocks
- Default value:
- None
- *Notes:
- Must not be0
- . 3600 (12 hours) is suggested.
- Standard Config Requirement:
- 3_600 base layer blocks (12 hours for an
- L2 on Ethereum, assuming 12 second L1 blocktime). This is an important value
- for constraining the sequencer's ability to re-order transactions; higher
- values would pose a risk to user protections.

channelTimeout

Maximum number of L1 blocks that a transaction channel frame can be considered valid. A transaction channel frame is a chunk of a compressed batch of transactions. After the timeout, the frame is dropped.

- Type:
- Number of blocks
- Default value:
- None
- Notes:
- 300 (1 hour) is suggested

p2pSequencerAddress

Address of the key that the Sequencer uses to sign blocks on the p2p network.

- Type:
- L1 Address
- Default value:
- None
- Notes:
- Sequencer, an address for which you own the private key

batchInboxAddress

Address that Sequencer transaction batches are sent to on L1.

- Type:
- L1 Address
- Default value:
- None
- Standard Config Requirement:
- Current convention is0xff000...000{chainId}
- .

batchSenderAddress

Address that nodes will filter for when searching for Sequencer transaction batches being sent to the batchInboxAddress. Can be updated later via the SystemConfig contract on L1.

- Type:
- L1 Address

- Default value:
- Batcher, an address for which you own the private key.
- Notes:
- Must not beaddress(0)

systemConfigStartBlock

SystemConfigStartBlock represents the block at which the op-node should start syncing from. It is an override to set this value on legacy networks where it is not set by default. It can be removed once all networks have this value set in their storage.

- Type:
- L2 Block Number
- Default value:
- None
- Standard Config Requirement:
- The block where the SystemConfig was
- initialized.

Chain Information

l1StartingBlockTag

Block tag for the L1 block where the L2 chain will begin syncing from. It is generally recommended to use a finalized block to avoid issues with reorgs.

- Type:
- Block hash
- Default value:
- None
- *Notes:
- Must not be0
- .

l1ChainID

Chain ID of the L1 chain.

- Type:
- Number
- Default value:
- None
- *Notes:
- Must not be0
- . 1 for L1 Ethereum mainnet, 11155111 for the
- Sepolia test network, and Seehere(opens in a new tab)
- for other blockchains.

l2ChainID

Chain ID of the L2 chain.

- Type:
- Number
- Default value:
- None
- *Notes:
- Must not be0
- . For security reasons, should be unique.
- Standard Config Requirement:
- Foundation-approved, globally unique value

l2GenesisBlockExtraData

L2GenesisBlockExtraData is configurable extradata. Will default to []byte("BEDROCK") if left unspecified.

- Type:
- Number

- Default value:
- []byte("BEDROCK")

superchainConfigGuardian

SuperchainConfigGuardian represents the GUARDIAN account in the SuperchainConfig. Has the ability to pause withdrawals.

- Type:
- L1 Address
- Default value:
- None
- Notes:
- Must not beaddress(0)

Gas

- Standard Config Requirement:
- Set such that Fee Margin is between 0 and
- 50%.
- Standard Config Requirement:
- No higher than 200_000_000 gas. Chain
- operators are driven to maintain a stable and reliable chain. When considering
- a change to this value, careful deliberation is necessary.

eip1559Elasticity

EIP1559Elasticity is the elasticity of the EIP1559 fee market.

- Type:
- Number
- Default value:
- None
- *Notes:
- Must not be0
- .

eip1559Denominator

EIP1559Denominator is the denominator of EIP1559 base fee market.

- Type:
- Number
- Default value:
- None
- *Notes:
- Must not be0
- .

eip1559DenominatorCanyon

EIP1559DenominatorCanyon is the denominator of EIP1559 base fee market when Canyon is active.

- Type:
- Number
- Default value:
- None
- *Notes:
- Must not be0
- if Canyon is activated.

gasPriceOracleBaseFeeScalar

GasPriceOracleBaseFeeScalar represents the value of the base fee scalar used for fee calculations.

- Type:
- Number
- Default value:

- None
- *Notes:
- Should not be0
- .

gasPriceOracleBlobBaseFeeScalar

GasPriceOracleBlobBaseFeeScalar represents the value of the blob base fee scalar used for fee calculations.

- Type:
- Number
- Default value:
- None
- *Notes:
- Should not be0
- .

Proposal Fields

These fields apply to output root proposals. The l2OutputOracleSubmissionInterval is configurable, see the section below for guidance.

l2OutputOracleStartingBlockNumber

Block number of the first OP Stack block. Typically this should be zero, but this may be non-zero for networks that have been upgraded from a legacy system (like OP Mainnet). Will be removed with the addition of permissionless proposals.

- Type:
- Number
- Default value:
- None
- *Notes:
- Should be0
- for new chains.

l2OutputOracleStartingTimestamp

Timestamp of the first OP Stack block. This MUST be the timestamp corresponding to the block defined by the l1StartingBlockTag. Will be removed with the addition of permissionless proposals.

- Type:
- Number
- Default value:
- None
- *Notes:
- his MUST be the timestamp corresponding to the block defined by
- the l1StartingBlockTag.

l2OutputOracleSubmissionInterval

Number of blocks between proposals to the L2OutputOracle. Will be removed with the addition of permissionless proposals.

- Type:
- Number of blocks
- Default value:
- None
- *Notes:
- Must not be0
- . 120 (4 minutes) is suggested.

finalizationPeriodSeconds

Number of seconds that a proposal must be available to challenge before it is considered finalized by the OptimismPortal contract.

- Type:
- Number of seconds
- Default value:

- None
- *Notes:
- Must not be0
- . Recommend 12 on test networks, seven days on
- production ones.
- Standard Config Requirement:
- 7 days. High security. Excessively safe
- upper bound that leaves enough time to consider social layer solutions to a
- hack if necessary. Allows enough time for other network participants to
- challenge the integrity of the corresponding output root.

l2OutputOracleProposer

Address that is allowed to submit output proposals to the L2OutputOracle contract. Will be removed when the OP Stack has permissionless proposals.

- Type:
- L1 Address
- Default value:
- None
- Notes:
- Must not beaddress(0)

l2OutputOracleChallenger

Address that is allowed to challenge output proposals submitted to the L2OutputOracle. Will be removed when the OP Stack has permissionless challenges.

- Type:
- L1 Address
- Default value:
- None
- Notes:
- Must not beaddress(0)
- . It is recommended to have a single admin
- address to retain a common security model.

Fee recipients

baseFeeVaultRecipient

BaseFeeVaultRecipient represents the recipient of fees accumulated in the BaseFeeVault. Can be an account on L1 or L2, depending on the BaseFeeVaultWithdrawalNetwork value.

- Type:
- L1 or L2 Address
- Default value:
- None
- Notes:
- Must not beaddress(0)
- . It is recommended to have a single admin
- address to retain a common security model.

l1FeeVaultRecipient

L1FeeVaultRecipient represents the recipient of fees accumulated in the L1FeeVault. Can be an account on L1 or L2, depending on the L1FeeVaultWithdrawalNetwork value.

- Type:
- L1 or L2 Address
- Default value:
- None
- Notes:
- Must not beaddress(0)
- . It is recommended to have a single admin
- address to retain a common security model.

sequencerFeeVaultRecipient

SequencerFeeVaultRecipient represents the recipient of fees accumulated in the SequencerFeeVault. Can be an account on L1 or L2, depending on the SequencerFeeVaultWithdrawalNetwork value.

- Type:
- L1 or L2 Address
- Default value:
- None
- Notes:
- Must not beaddress(0)
- . It is recommended to have a single admin
- address to retain a common security model.

Minimum Fee Withdrawal Amounts

Withdrawals to L1 are expensive and the minimum fee is to prevent overhead costs of continuous tiny withdrawals. If the withdrawal execution costs more than the fee-reward, then the fee Must not be collected economically.

baseFeeVaultMinimumWithdrawalAmount

BaseFeeVaultMinimumWithdrawalAmount represents the minimum withdrawal amount for the BaseFeeVault.

- Type:
- Number in wei
- Default value:
- None

l1FeeVaultMinimumWithdrawalAmount

L1FeeVaultMinimumWithdrawalAmount represents the minimum withdrawal amount for the L1FeeVault.

- Type:
- Number in wei
- Default value:
- None

sequencerFeeVaultWithdrawalAmount

SequencerFeeVaultMinimumWithdrawalAmount represents the minimum withdrawal amount for the SequencerFeeVault.

- Type:
- Number in wei
- Default value:
- None

Withdrawal Network

baseFeeVaultWithdrawalNetwork

BaseFeeVaultWithdrawalNetwork represents the withdrawal network for the BaseFeeVault. value of 0 will withdraw ETH to the recipient address on L1 and a value of 1 will withdraw ETH to the recipient address on L2.

- Type:
- Number representing network enum
- Default value:
- None
- Notes:
- Withdrawals to Ethereum are more expensive.

l1FeeVaultWithdrawalNetwork

L1FeeVaultWithdrawalNetwork represents the withdrawal network for the L1FeeVault. A value of 0 will withdraw ETH to the recipient address on L1 and a value of 1 will withdraw ETH to the recipient address on L2.

- Type:
- Number representing network enum
- Default value:
- None
- Notes:

- Withdrawals to Ethereum are more expensive.

sequencerFeeVaultWithdrawalNetwork

SequencerFeeVaultWithdrawalNetwork represents the withdrawal network for the SequencerFeeVault. A value of 0 will withdraw ETH to the recipient address on L1 and a value of 1 will withdraw ETH to the recipient address on L2.

- Type:
- Number representing network enum
- Default value:
- None
- Notes:
- Withdrawals to Ethereum are more expensive.

Fault Proofs

faultGameAbsolutePrestate

FaultGameAbsolutePrestate is the absolute prestate of Cannon. This is computed by generating a proof from the 0th -> 1st instruction and grabbing the prestate from the output JSON. All honest challengers should agree on the setup state of the program.

- Type:
- Hash
- Default value:
- None

faultGameMaxDepth

FaultGameMaxDepth is the maximum depth of the position tree within the fault dispute game.2^{FaultGameMaxDepth} is how many instructions the execution trace bisection game supports. Ideally, this should be conservatively set so that there is always enough room for a full Cannon trace.

- Type:
- Number
- Default value:
- None

faultGameClockExtension

FaultGameClockExtension is the amount of time that the dispute game will set the potential grandchild claim's, clock to, if the remaining time is less than this value at the time of a claim's creation.

- Type:
- Number
- Default value:
- None

faultGameMaxClockDuration

FaultGameMaxClockDuration is the maximum amount of time that may accumulate on a team's chess clock before they may no longer respond.

- Type:
- Number
- Default value:
- None

faultGameGenesisBlock

FaultGameGenesisBlock is the block number for genesis.

- Type:
- Number
- Default value:
- None

faultGameGenesisOutputRoot

FaultGameGenesisOutputRoot is the output root for the genesis block.

- Type:
- Hash
- Default value:
- None

faultGameSplitDepth

FaultGameSplitDepth is the depth at which the fault dispute game splits from output roots to execution trace claims.

- Type:
- Number
- Default value:
- None

faultGameWithdrawalDelay

FaultGameWithdrawalDelay is the number of seconds that users must wait before withdrawing ETH from a fault game.

- Type:
- Number
- Default value:
- None

preimageOracleMinProposalSize

PreimageOracleMinProposalSize is the minimum number of bytes that a large preimage oracle proposal can be.

- Type:
- Number
- Default value:
- None

preimageOracleChallengePeriod

PreimageOracleChallengePeriod is the number of seconds that challengers have to challenge a large preimage proposal.

- Type:
- Number of Seconds
- Default value:
- None

proofMaturityDelaySeconds

ProofMaturityDelaySeconds is the number of seconds that a proof must be mature before it can be used to finalize a withdrawal.

- Type:
- Number
- Default value:
- None
- *Notes:
- Should not be0
- .

disputeGameFinalityDelaySeconds

DisputeGameFinalityDelaySeconds is an additional number of seconds a dispute game must wait before it can be used to finalize a withdrawal.

- Type:
- Number
- Default value:
- None
- *Notes:
- Should not be0
- .

respectedGameType

RespectedGameType is the dispute game type that the OptimismPortal contract will respect for finalizing withdrawals.

- Type:
- Number
- Default value:
- None

useFaultProofs

UseFaultProofs is a flag that indicates if the system is using fault proofs instead of the older output oracle mechanism.

- Type:
- Boolean
- Default value:
- None
- *Notes:
- You should understand the implications of running a Fault Proof
- chain.

Custom Gas Token

useCustomGasToken

UseCustomGasToken is a flag to indicate that a custom gas token should be used.

- Type:
- boolean
- Default value:
- None

customGasTokenAddress

CustomGasTokenAddress is the address of the ERC20 token to be used to pay for gas on L2.

- Type:
- Address
- Default value:
- None
- Notes:
- Must not beaddress(0)
- .

Alt-DA Mode / Plasma Mode

usePlasma

UsePlasma is a flag that indicates if the system is using op-plasma

- Type:
- bool
- Default value:
- None
- Notes:

daCommitmentType

DACommitmentType specifies the allowed commitment

- Type:
- string
- Default value:
- None
- Notes:
- DACommitmentType must be either KeccakCommitment or
- GenericCommitment

daChallengeWindow

DAChallengeWindow represents the block interval during which the availability of a data commitment can be challenged.

- Type:
- Number
- Default value:
- None
- Notes:
- DAChallengeWindow must not be 0 when using plasma mode

daResolveWindow

DAResolveWindow represents the block interval during which a data availability challenge can be resolved.

- Type:
- Number
- Default value:
- None
- Notes:
- DAChallengeWindow must no be 0 when using plasma mode

daBondSize

DABondSize represents the required bond size to initiate a data availability challenge.

- Type:
- Number
- Default value:
- None

daResolverRefundPercentage

DAResolverRefundPercentage represents the percentage of the resolving cost to be refunded to the resolver such as 100 means 100% refund.

- Type:
- Number
- Default value:
- None

daChallengeProxy

DAChallengeProxy represents the L1 address of the DataAvailabilityChallenge contract.

- Type:
- Address
- Default value:
- None
- Notes:
- Must not be address(0) when using plasma mode

Interoperability

useInterop

UseInterop is a flag that indicates if the system is using interop.

- Type:
- boolean
- Default value:
- None

Governance

enableGovernance

EnableGovernance determines whether to include governance token predeploy.

- Type:
- boolean
- Default value:
- None

governanceTokenSymbol

GovernanceTokenSymbol represents the ERC20 symbol of the GovernanceToken.

- Type
- : string
- Default value:
- None

governanceTokenName

GovernanceTokenName represents the ERC20 name of the GovernanceToken

- Type
- : string
- Default value:
- None

governanceTokenOwner

GovernanceTokenOwner represents the owner of the GovernanceToken. Has the ability to mint and burn tokens.

- Type
- : L2 Address
- Default value:
- None

Miscellaneous

fundDevAccounts

FundDevAccounts determines whether to fund the dev accounts. Should only be used during devnet deployments.

- Type
- : Boolean

requiredProtocolVersion

RequiredProtocolVersion indicates the protocol version that nodes are recommended to adopt, to stay in sync with the network.

- Type
- : String

recommendedProtocolVersion

RecommendedProtocolVersion indicates the protocol version that nodes are recommended to adopt, to stay in sync with the network.

- Type
- : String

Deprecated

(DEPRECATED

) gasPriceOracleScalar

GasPriceOracleScalar represents the initial value of the gas scalar in the GasPriceOracle predeploy. Deprecated: Since Ecotone, this field is superseded by GasPriceOracleBaseFeeScalar and GasPriceOracleBlobBaseFeeScalar.

(DEPRECATED

) gasPriceOracleOverhead

GasPriceOracleOverhead represents the initial value of the gas overhead in the GasPriceOracle predeploy. Deprecated: Since Ecotone, this field is superseded by GasPriceOracleBaseFeeScalar and GasPriceOracleBlobBaseFeeScalar.

(DEPRECATED

) deploymentWaitConfirmations

DeploymentWaitConfirmations is the number of confirmations to wait during deployment. This is DEPRECATED and should be removed in a future PR.

(DEPRECATED

) numDeployConfirmations

Number of confirmations to wait when deploying smart contracts to L1.

Overview Batcher Configuration