

# Backend ENVs: Chain-Specific

ENVS for specific chains and rollups

## Chain Specific ENVs

The following ENVs are specific to different types of chains and rollups. To use these specific variables you will also need to set the CHAIN\_TYPE variable to the appropriate chain.

Variable Description Parameters CHAIN\_TYPE Specifies the model of data and enables fetchers specific to the chain. Available values are ethereum , optimism , arbitrum , polygon\_zkevm , polygon\_edge , rsk , shibarium , stability , suave , zetachain , filecoin , default . Needs Recompile: ☒ Version: v5.3.0+ Default: default Applications: API, Indexer

## Ethereum Management

INDEXER\_BEACON\_RPC\_URL Beacon Chain RPC endpoint used to fetch blob sidecars. Required if INDEXER\_OPTIMISM\_L1\_BATCH\_START\_BLOCK is not empty. Implemented in [#9168](#) . Version: v.6.2.0+ Default: http://localhost:5052 Applications: Indexer INDEXER\_DISABLE\_BEACON\_BLOB\_FETCHER If true the Beacon data blobs fetcher won't be started, new transaction and block fields will still be extracted. Implemented in [#9168](#) . Version: v.6.2.0+ Default: false Applications: Indexer INDEXER\_BEACON\_BLOB\_FETCHER\_SLOT\_DURATION Slot duration in the Beacon Chain in seconds. Required if INDEXER\_OPTIMISM\_L1\_BATCH\_START\_BLOCK is not empty. Implemented in [#9168](#) . Version: v.6.2.0+ Default: 12 Applications: Indexer INDEXER\_BEACON\_BLOB\_FETCHER\_REFERENCE\_SLOT Any past finalized Beacon Chain slot number. Used as reference for blob inclusion slot calculations. Required if INDEXER\_OPTIMISM\_L1\_BATCH\_START\_BLOCK is not empty. Implemented in [#9168](#) . Version: v.6.2.0+ Default: 8000000 Applications: Indexer INDEXER\_BEACON\_BLOB\_FETCHER\_REFERENCE\_TIMESTAMP UTC timestamp of the Beacon Chain slot specified in INDEXER\_BEACON\_BLOB\_FETCHER\_REFERENCE\_SLOT . Used as reference for blob inclusion slot calculations. Required if INDEXER\_OPTIMISM\_L1\_BATCH\_START\_BLOCK is not empty. Implemented in [#9168](#) . Version: v.6.2.0+ Default: 1702824023 Applications: Indexer INDEXER\_BEACON\_BLOB\_FETCHER\_START\_BLOCK Beacon Chain blob fetcher start block. On start-up, indexer will only look for missed blobs beyond this block number. It's recommended to set this block to the first block after the Dencun hardfork. Implemented in [#9168](#) . Version: v.6.2.0+ Default: 19200000 Applications: Indexer INDEXER\_BEACON\_BLOB\_FETCHER\_END\_BLOCK Beacon Chain blob fetcher end block. On start-up, indexer will only look for missed blobs before this block number. If set to 0 , then all recent till latest will be traversed. Implemented in [#9168](#) . Version: v.6.2.0+ Default: 0 Applications: Indexer

## Polygon Edge Management

Polygon edge is no longer supported by the Polygon team in favor of the Polygon CDK [more info](#)

INDEXER\_POLYGON\_EDGE\_L1\_RPC The RPC endpoint for L1 used to fetch deposit or withdrawal events. Implemented in [#8180](#) . Version: v.5.3.0+ Default: (empty) Applications: Indexer

INDEXER\_POLYGON\_EDGE\_L1\_EXIT\_HELPER\_CONTRACT The address of ExitHelper contract on L1 (root chain) used to fetch withdrawal exits. Required for withdrawal events indexing. Implemented in [#8180](#) . Version: v.5.3.0+ Default: (empty) Applications: Indexer

INDEXER\_POLYGON\_EDGE\_L1\_WITHDRAWALS\_START\_BLOCK The number of start block on L1 (root chain) to index withdrawal exits. If the table of withdrawal exits is not empty, the process will continue indexing from the last indexed message. If empty or not defined, the withdrawal exits are not indexed. Implemented in [#8180](#) . Version: v.5.3.0+ Default: (empty) Applications: Indexer

INDEXER\_POLYGON\_EDGE\_L1\_STATE\_SENDER\_CONTRACT The address of StateSender contract on L1 (root chain) used to fetch deposits. Required for deposit events indexing. Implemented in [#8180](#) . Version: v.5.3.0+ Default: (empty) Applications: Indexer

INDEXER\_POLYGON\_EDGE\_L1\_DEPOSITS\_START\_BLOCK The number of start block on L1 (root chain) to index deposits. If the table of deposits is not empty, the process will continue indexing from the last indexed message. If empty or not defined, the deposits are not indexed. Implemented in [#8180](#) . Version: v.5.3.0+ Default: (empty) Applications: Indexer

INDEXER\_POLYGON\_EDGE\_L2\_STATE\_SENDER\_CONTRACT The address of L2StateSender contract on L2 (child chain) used to fetch withdrawals. Required for withdrawal events indexing. Implemented in [#8180](#) . Version: v.5.3.0+ Default: (empty) Applications: Indexer

INDEXER\_POLYGON\_EDGE\_L2\_WITHDRAWALS\_START\_BLOCK The number of start block on L2 (child chain) to index withdrawals. If the table of withdrawals is not empty, the process will fill gaps and then continue indexing from the last indexed message. If empty or not defined, the withdrawals are not indexed. Implemented in [#8180](#) . Version: v.5.3.0+ Default: (empty) Applications: Indexer

INDEXER\_POLYGON\_EDGE\_L2\_STATE\_RECEIVER\_CONTRACT The address of StateReceiver contract on L2 (child chain) used to fetch deposit executes. Required for deposit events indexing. Implemented in [#8180](#) . Version: v.5.3.0+ Default: (empty) Applications: Indexer

INDEXER\_POLYGON\_EDGE\_L2\_DEPOSITS\_START\_BLOCK The number of start block on L2 (child chain) to index deposit executes. If the table of deposit executes is not empty, the process will fill gaps and then continue indexing from the last indexed message. If empty or not defined, the deposit executes are not indexed. Implemented in [#8180](#) . Version: v.5.3.0+ Default: (empty) Applications: Indexer

INDEXER\_POLYGON\_EDGE\_ETH\_GET\_LOGS\_RANGE\_SIZE Block range size for eth\_getLogs request in Polygon Edge indexer modules. Implemented in [#8180](#) . Version: v.5.3.0+ Default: (empty) Applications: Indexer

## Rootstock management

ROOTSTOCK\_REMASC\_ADDRESS The address hash of remasc address on Rootstock chain. Implemented in [#8542](#) . Version: v.5.3.0+ Default: (empty) Applications: API  
 ROOTSTOCK\_BRIDGE\_ADDRESS The address hash of bridge address on Rootstock chain. Implemented in [#8542](#) . Version: v.5.3.0+ Default: (empty) Applications: API  
 INDEXER\_DISABLE\_ROOTSTOCK\_DATA\_FETCHER If true the fetcher of Rootstock specific fields for blocks that are already in the database won't be started, fields from new blocks will be extracted. Implemented in [#8742](#) . Version: v.5.3.2+ Default: (empty) Applications: Indexer  
 INDEXER\_ROOTSTOCK\_DATA\_FETCHER\_INTERVAL The interval between fetching the next INDEXER\_ROOTSTOCK\_DATA\_FETCHER\_DB\_BATCH\_SIZE blocks from the database and the node, used to configure the load on the database and JSON-RPC node. Implemented in [#8742](#) . Version: v.5.3.2+ Default: (empty) Applications: Indexer  
 INDEXER\_ROOTSTOCK\_DATA\_FETCHER\_BATCH\_SIZE The number of requests in one JSON-RPC batch request, used to configure the load or RPS on JSON-RPC node. Implemented in [#8742](#) . Version: v.5.3.2+ Default: (empty) Applications: Indexer  
 INDEXER\_ROOTSTOCK\_DATA\_FETCHER\_CONCURRENCY The number of simultaneous requests to the JSON-RPC node, used to configure the load or RPS on JSON-RPC node. Implemented in [#8742](#) . Version: v.5.3.2+ Default: (empty) Applications: Indexer  
 INDEXER\_ROOTSTOCK\_DATA\_FETCHER\_DB\_BATCH\_SIZE The number of blocks fetched from the database in one database query. Implemented in [#8742](#) . Version: v.5.3.2+ Default: (empty) Applications: Indexer

## Shibarium Management

INDEXER\_SHIBARIUM\_L1\_RPC The RPC endpoint for L1 used to fetch deposit or withdrawal events. Implemented in [#8929](#) . Version: v.6.1.0+ Default: (empty) Applications: Indexer  
 INDEXER\_SHIBARIUM\_L1\_START\_BLOCK The number of start block on L1 to index L1 events. If the table of bridge operations is not empty, the process will continue indexing from the last indexed L1 event. If empty or not defined, the L1 events are not handled. Implemented in [#8929](#) . Version: v.6.1.0+ Default: (empty) Applications: Indexer  
 INDEXER\_SHIBARIUM\_L1\_DEPOSIT\_MANAGER\_CONTRACT The address of DepositManagerProxy contract on L1 used to fetch BONE token deposits. Required for L1 events indexing. Implemented in [#8929](#) . Version: v.6.1.0+ Default: (empty) Applications: Indexer  
 INDEXER\_SHIBARIUM\_L1\_ETHER\_PREDICATE\_CONTRACT The address of EtherPredicateProxy contract on L1 used to fetch ETH deposits and withdrawals. Required for L1 events indexing. Implemented in [#8929](#) . Version: v.6.1.0+ Default: (empty) Applications: Indexer  
 INDEXER\_SHIBARIUM\_L1\_ERC20\_PREDICATE\_CONTRACT The address of ERC20PredicateProxy contract on L1 used to fetch ERC20 token deposits and withdrawals. Required for L1 events indexing. Implemented in [#8929](#) . Version: v.6.1.0+ Default: (empty) Applications: Indexer  
 INDEXER\_SHIBARIUM\_L1\_ERC721\_PREDICATE\_CONTRACT The address of ERC721PredicateProxy contract on L1 used to fetch ERC721 token deposits and withdrawals. Optional for L1 events indexing. Implemented in [#8929](#) . Version: v.6.1.0+ Default: (empty) Applications: Indexer  
 INDEXER\_SHIBARIUM\_L1\_ERC1155\_PREDICATE\_CONTRACT The address of ERC1155PredicateProxy contract on L1 used to fetch ERC1155 token deposits and withdrawals. Optional for L1 events indexing. Implemented in [#8929](#) . Version: v.6.1.0+ Default: (empty) Applications: Indexer  
 INDEXER\_SHIBARIUM\_L1\_WITHDRAW\_MANAGER\_CONTRACT The address of WithdrawManagerProxy contract on L1 used to fetch BONE token withdrawals. Required for L1 events indexing. Implemented in [#8929](#) . Version: v.6.1.0+ Default: (empty) Applications: Indexer  
 INDEXER\_SHIBARIUM\_L2\_START\_BLOCK The number of start block on L2 to index L2 events. If the table of bridge operations is not empty, the process will continue indexing from the last indexed L2 event. If empty or not defined, the L2 events are not handled. Implemented in [#8929](#) . Version: v.6.1.0+ Default: (empty) Applications: Indexer  
 INDEXER\_SHIBARIUM\_L2\_CHILD\_CHAIN\_CONTRACT The address of ChildChain contract on L2 used to fetch BONE token deposits. Required for L2 events indexing. Implemented in [#8929](#) . Version: v.6.1.0+ Default: (empty) Applications: Indexer  
 INDEXER\_SHIBARIUM\_L2\_WETH\_CONTRACT The address of WETH contract on L2 used to fetch ETH deposits and withdrawals. Required for L2 events indexing. Implemented in [#8929](#) . Version: v.6.1.0+ Default: (empty) Applications: Indexer  
 INDEXER\_SHIBARIUM\_L2\_BONE\_WITHDRAW\_CONTRACT The address of a contract which emits Withdraw event on L2. Used to fetch BONE token withdrawals. Required for L2 events indexing. Implemented in [#8929](#) . Version: v.6.1.0+ Default: (empty) Applications: Indexer

## Polygon zkEVM Rollup management

INDEXER\_POLYGON\_ZKEVM\_BATCHES\_ENABLED Enables Polygon zkEVM batches fetcher. Implemented in [#7584](#) . Version: v.5.3.1+ Default: false Applications: Indexer  
 INDEXER\_POLYGON\_ZKEVM\_BATCHES\_CHUNK\_SIZE The number of Polygon zkEVM batches in one chunk when reading them from RPC. Implemented in [#7584](#) . Version: v.5.3.1+ Default: 20 Applications: Indexer  
 INDEXER\_POLYGON\_ZKEVM\_BATCHES\_RECHECK\_INTERVAL The latest batch rechecking interval, seconds. Implemented in [#7584](#) . Version: v.5.3.1+ Default: 60 Applications: Indexer  
 INDEXER\_POLYGON\_ZKEVM\_L1\_RPC The RPC endpoint for L1 used to fetch Deposit or Withdrawal bridge events. Implemented in [#9098](#) . Version: v.6.2.0+ Default: (empty) Applications: Indexer  
 INDEXER\_POLYGON\_ZKEVM\_L1\_BRIDGE\_START\_BLOCK The number of a start block on L1 to index L1 bridge events. If the table of bridge operations is not empty, the process will continue indexing from the last indexed L1 event. If empty or not defined, the L1 events are not handled. Implemented in [#9098](#) . Version: v.6.2.0+ Default: (empty) Applications: Indexer  
 INDEXER\_POLYGON\_ZKEVM\_L1\_BRIDGE\_CONTRACT The address of PolygonZkEVMBridgeV2 contract on L1 used to fetch L1 bridge events. Required for L1 bridge events indexing. Implemented in [#9098](#) . Version: v.6.2.0+ Default: (empty) Applications: Indexer  
 INDEXER\_POLYGON\_ZKEVM\_L1\_BRIDGE\_NETWORK\_ID L1 Network ID in terms of Polygon zkEVM bridge (0 = Ethereum Mainnet, 1 = Polygon zkEVM, 2 = Astar zkEVM, etc.). Required if INDEXER\_POLYGON\_ZKEVM\_L1\_BRIDGE\_START\_BLOCK or INDEXER\_POLYGON\_ZKEVM\_L2\_BRIDGE\_START\_BLOCK is defined. Implemented in [#9637](#) . Version: v.6.4.0+ Default: (empty) Applications: Indexer  
 INDEXER\_POLYGON\_ZKEVM\_L1\_BRIDGE\_ROLLUP\_INDEX L1 Rollup index in terms of Polygon zkEVM bridge (0 = Polygon zkEVM, 1 = Astar zkEVM, etc.). Not defined if L1 is Ethereum Mainnet. Required if L1 is not Ethereum Mainnet and INDEXER\_POLYGON\_ZKEVM\_L1\_BRIDGE\_START\_BLOCK

or `INDEXER_POLYGON_ZKEVM_L2_BRIDGE_START_BLOCK` is defined. Implemented in [#9637](#) . Version: v.6.4.0+ Default: (empty) Applications: Indexer `INDEXER_POLYGON_ZKEVM_L1_BRIDGE_NATIVE_SYMBOL` The symbol of the native coin on L1 to display it in the table of the bridge Deposits and Withdrawals on UI. Implemented in [#9098](#) . Version: v.6.2.0+ Default:ETH Applications: Indexer `INDEXER_POLYGON_ZKEVM_L1_BRIDGE_NATIVE_DECIMALS` The number of decimals to correctly display an amount of native coins for some Deposit or Withdrawal bridge operations on UI. Implemented in [#9098](#) . Version: v.6.2.0+ Default:18 Applications: Indexer `INDEXER_POLYGON_ZKEVM_L2_BRIDGE_START_BLOCK` The number of a start block on L2 to index L2 bridge events. If the table of bridge operations is not empty, the process will continue indexing from the last indexed L2 event. If empty or not defined, the L2 events are not handled. Implemented in [#9098](#) . Version: v.6.2.0+ Default: (empty) Applications: Indexer `INDEXER_POLYGON_ZKEVM_L2_BRIDGE_CONTRACT` The address of PolygonZkEVMBridgeV2 contract on L2 used to fetch L2 bridge events. Required for L2 bridge events indexing. Implemented in [#9098](#) . Version: v.6.2.0+ Default: (empty) Applications: Indexer `INDEXER_POLYGON_ZKEVM_L2_BRIDGE_NETWORK_ID` L2 Network ID in terms of Polygon zkEVM bridge (1 = Polygon zkEVM, 2 = Astar zkEVM, etc.). Required if `INDEXER_POLYGON_ZKEVM_L1_BRIDGE_START_BLOCK` or `INDEXER_POLYGON_ZKEVM_L2_BRIDGE_START_BLOCK` is defined. Implemented in [#9637](#) . Version: v.6.4.0+ Default: (empty) Applications: Indexer `INDEXER_POLYGON_ZKEVM_L2_BRIDGE_ROLLUP_INDEX` L2 Rollup index in terms of Polygon zkEVM bridge (0 = Polygon zkEVM, 1 = Astar zkEVM, etc.). Required if `INDEXER_POLYGON_ZKEVM_L1_BRIDGE_START_BLOCK` or `INDEXER_POLYGON_ZKEVM_L2_BRIDGE_START_BLOCK` is defined. Implemented in [#9637](#) . Version: v.6.4.0+ Default: (empty) Applications: Indexer

## Optimism Rollup Management

Supported together with ENV setting `CHAIN_TYPE=optimism` `INDEXER_OPTIMISM_L1_RPC` The RPC endpoint for L1 used to fetch transaction batches, output roots, deposits, or withdrawal events. Implemented in [#6980](#) . Version: v.6.3.0+ Default: (empty) Applications: Indexer `INDEXER_OPTIMISM_L1_PORTAL_CONTRACT` The address of OptimismPortal contract on L1 used to fetch deposits and withdrawal events. Required for deposits and withdrawal events indexing. Implemented in [#6980](#) . Version: v.6.3.0+ Default: (empty) Applications: Indexer `INDEXER_OPTIMISM_L1_BATCH_START_BLOCK` The number of start block on L1 to index transaction batches. If the table of batches is not empty, the process will continue indexing from the last indexed batch. Implemented in [#6980](#) . Version: v.6.3.0+ Default: (empty) Applications: Indexer `INDEXER_OPTIMISM_L1_BATCH_INBOX` The inbox address to index transaction batches on L1. Required if `INDEXER_OPTIMISM_L1_BATCH_START_BLOCK` is not empty. Implemented in [#6980](#) . Version: v.6.3.0+ Default: (empty) Applications: Indexer `INDEXER_OPTIMISM_L1_BATCH_SUBMITTER` The batch submitter address to index transaction batches on L1. Required if `INDEXER_OPTIMISM_L1_BATCH_START_BLOCK` is not empty. Implemented in [#6980](#) . Version: v.6.3.0+ Default: (empty) Applications: Indexer `INDEXER_OPTIMISM_L1_BATCH_BLOCKSCOUT_BLOBS_API_URL` Defines a URL to Blockscout Blobs API to retrieve L1 blobs from that. Example for Sepolia: <https://eth-sepolia.blockscout.com/api/v2/blobs> . Required if `INDEXER_OPTIMISM_L1_BATCH_START_BLOCK` is not empty. Implemented in [#9571](#) . Version: v.6.3.0+ Default: (empty) Applications: Indexer `INDEXER_OPTIMISM_L1_BATCH_BLOCKS_CHUNK_SIZE` Blocks chunk size to send batch RPC requests. Implemented in [#6980](#) . Version: v.6.3.0+ Default:4 Applications: Indexer `INDEXER_OPTIMISM_L2_BATCH_GENESIS_BLOCK_NUMBER` L2 genesis block number for Optimism chain. Required if `INDEXER_OPTIMISM_L1_BATCH_START_BLOCK` is defined. The block number can be found at [superchain registry](#) . [Example](#) . Implemented in [#9260](#) . Version: v.6.3.0+ Default: (empty) Applications: Indexer `INDEXER_OPTIMISM_L1_OUTPUT_ROOTS_START_BLOCK` The number of start block on L1 to index output roots. If the table of output roots is not empty, the process will continue indexing from the last indexed root. Implemented in [#6980](#) . Version: v.6.3.0+ Default: (empty) Applications: Indexer `INDEXER_OPTIMISM_L1_OUTPUT_ORACLE_CONTRACT` The address of OutputOracle contract on L1 used to fetch output roots. Required for output roots indexing when `INDEXER_OPTIMISM_L1_OUTPUT_ROOTS_START_BLOCK` is not empty. Implemented in [#6980](#) . Version: v.6.3.0+ Default: (empty) Applications: Indexer `INDEXER_OPTIMISM_L1_DEPOSITS_START_BLOCK` The number of the L1 block from which deposits will be fetched. Implemented in [#6993](#) . Version: v.6.3.0+ Default: (empty) Applications: Indexer `INDEXER_OPTIMISM_L1_DEPOSITS_BATCH_SIZE` Number of blocks in a single `eth_getLogs` request. Implemented in [#6993](#) . Version: v.6.3.0+ Default:500 Applications: Indexer `INDEXER_OPTIMISM_L1_WITHDRAWALS_START_BLOCK` The number of start block on L1 to index withdrawal events. If the table of withdrawal events is not empty, the process will continue indexing from the last indexed withdrawal event. Implemented in [#6980](#) . Version: v.6.3.0+ Default: (empty) Applications: Indexer `INDEXER_OPTIMISM_L2_WITHDRAWALS_START_BLOCK` The number of start block on L2 to index withdrawals. If the table of withdrawals is not empty, the process will continue indexing from the last indexed withdrawal. Implemented in [#6980](#) . Version: v.6.3.0+ Default: (empty) Applications: Indexer `INDEXER_OPTIMISM_L2_MESSAGE_PASSER_CONTRACT` The address of L2ToL1MessagePasser contract to index withdrawals. Required if `INDEXER_OPTIMISM_L2_WITHDRAWALS_START_BLOCK` is not empty. Implemented in [#6980](#) . Version: v.6.3.0+ Default: (empty) Applications: Indexer

## zkSync Rollup Management

Supported together with setting `CHAIN_TYPE=zkSync` `INDEXER_ZKSYNC_BATCHES_ENABLED` Enables Polygon zkEVM batches fetcher. Implemented in [#9080](#) . Version: v.6.3.0+ Default:false Applications: Indexer `INDEXER_ZKSYNC_BATCHES_CHUNK_SIZE` The number of RPC calls in one request when reading data from RPC. Implemented in [#9080](#) . Version: v.6.3.0+ Default:50 Applications: Indexer `INDEXER_ZKSYNC_NEW_BATCHES_MAX_RANGE` Maximum amount of batches requested if Blockscout does not have all batches synced. Implemented in [#9080](#) . Version: v.6.3.0+ Default:50 Applications: Indexer



INDEXER\_ZKSYNC\_NEW\_BATCHES\_RECHECK\_INTERVAL The latest batch rechecking interval, seconds. Implemented in [#9080](#) . Version: v.6.3.0+ Default:60 Applications: Indexer INDEXER\_ZKSYNC\_L1\_RPC The RPC endpoint for L1 used to fetch status of batches. Implemented in [#9080](#) . Version: v.6.3.0+ Default: (empty) Applications: Indexer INDEXER\_ZKSYNC\_BATCHES\_STATUS\_RECHECK\_INTERVAL The batches status rechecking interval, seconds. Implemented in [#9080](#) . Version: v.6.3.0+ Default:50 Applications: Indexer

## Arbitrum Rollup Management

Supported together with setting CHAIN\_TYPE=arbitrum INDEXER\_ARBITRUM\_ARBSYS\_CONTRACT The address of ArbSys contract on the rollup. Implemented in [#9312](#) . Version: v.TBD Default:0x00 Applications: Indexer INDEXER\_ARBITRUM\_L1\_RPC The RPC endpoint for L1 used to fetch transaction batches, block confirmations, and cross-chain messages. Implemented in [#9312](#) . Version: v.TBD Default: empty Applications: Indexer INDEXER\_ARBITRUM\_L1\_RPC\_CHUNK\_SIZE The number of RPC calls in one request when reading data from RPC. Implemented in [#9312](#) . Version: v.TBD Default:20 Applications: Indexer INDEXER\_ARBITRUM\_L1\_RPC\_HISTORICAL\_BLOCKS\_RANGE The block range size for the eth\_getLogs request. Implemented in [#9312](#) . Version: v.TBD Default:1000 Applications: Indexer INDEXER\_ARBITRUM\_L1\_ROLLUP\_CONTRACT The Arbitrum Rollup contract address on L1. Implemented in [#9312](#) . Version: v.TBD Default: empty Applications: Indexer INDEXER\_ARBITRUM\_L1\_ROLLUP\_INIT\_BLOCK The block number where the Arbitrum rollup contract has been deployed. Implemented in [#9312](#) . Version: v.TBD Default:1 Applications: Indexer INDEXER\_ARBITRUM\_L1\_COMMON\_START\_BLOCK The block number at which synchronization will commence. It will proceed in both directions: discovering new batches up to the chain head, and historical batches until INDEXER\_ARBITRUM\_L1\_ROLLUP\_INIT\_BLOCK . If omitted, the synchronization starts from the latest block. Implemented in [#9312](#) . Version: v.TBD Default:latest block Applications: Indexer INDEXER\_ARBITRUM\_ROLLUP\_CHUNK\_SIZE The number of RPC calls in one request when reading data from RPC. Implemented in [#9312](#) . Version: v.TBD Default:20 Applications: Indexer INDEXER\_ARBITRUM\_BRIDGE\_MESSAGES\_TRACKING\_ENABLED Enables monitoring on L1 of cross-chain messages directed to L2 and catchup on L2 of historical cross-chain messages. Implemented in [#9312](#) . Version: v.TBD Default: (empty) Applications: Indexer INDEXER\_ARBITRUM\_MISSED\_MESSAGES\_RECHECK\_INTERVAL Interval to re-check on the rollup historical messages directed to and from the rollup, in seconds. Implemented in [#9312](#) . Version: v.TBD Default:3600 Applications: Indexer INDEXER\_ARBITRUM\_MISSED\_MESSAGES\_TO\_L2\_BLOCK\_DEPTH Amount of L2 blocks to revisit to identify historical L1-to-L2 messages in the messages catchup task by handling block transactions. Implemented in [#9312](#) . Version: v.TBD Default:50 Applications: Indexer INDEXER\_ARBITRUM\_MISSED\_MESSAGES\_TO\_L1\_BLOCK\_DEPTH Amount of L2 blocks to revisit to identify historical L2-to-L1 messages in the messages catchup task by handling logs in receipts. Implemented in [#9312](#) . Version: v.TBD Default:1000 Applications: Indexer INDEXER\_ARBITRUM\_TRACKING\_MESSAGES\_ON\_L1\_RECHECK\_INTERVAL Interval to re-check on L1 new messages directed to the rollup, in seconds. Implemented in [#9312](#) . Version: v.TBD Default:20 Applications: Indexer INDEXER\_ARBITRUM\_BATCHES\_TRACKING\_ENABLED Enables monitoring of transaction batches, block confirmations, and L2-to-L1 messages executions. Implemented in [#9312](#) . Version: v.TBD Default: (empty) Applications: Indexer INDEXER\_ARBITRUM\_BATCHES\_TRACKING\_RECHECK\_INTERVAL Interval to re-check on L1 new and historical batches, confirmations, and executions, in seconds. Implemented in [#9312](#) . Version: v.TBD Default:20 Applications: Indexer INDEXER\_ARBITRUM\_BATCHES\_TRACKING\_L1\_FINALIZATION\_CHECK\_ENABLED Whether L1 transactions related to batches, confirmations, and executions need to be monitored for finalization or not. Implemented in [#9312](#) . Version: v.TBD Default:false Applications: Indexer INDEXER\_ARBITRUM\_BATCHES\_TRACKING\_MESSAGES\_TO\_BLOCKS\_SHIFT Difference between the message count and actual rollup block numbers in the SequencerBatchDelivered event on L1. Applicable for ArbitrumOne only. Implemented in [#9312](#) . Version: v.TBD Default:0 Applications: Indexer INDEXER\_ARBITRUM\_CONFIRMATIONS\_TRACKING\_FINALIZED Whether to choose safe (true) or latest (false) block to discover new confirmations. This setting is required to reduce latency between the actual confirmation transaction and its discovery. Recommended to have true on rollups which use Ethereum Mainnet as L1. Determines how fast new confirmations will be discovered. Implemented in [#9312](#) . Version: v.TBD Default:true Applications: Indexer INDEXER\_ARBITRUM\_NEW\_BATCHES\_LIMIT The number of batches to be handled and imported at once. This is applicable for cases when dozens of batches are found in one discovery iteration, and it is necessary to import them in chunks to avoid the entire process being aborted if any errors occur. Implemented in [#9312](#) . Version: v.TBD Default:10 Applications: Indexer

Last updated 1 day ago