NodeInterface reference

The Arbitrum Nitro software includes a specialNodeInterface contract available at address0xc8 that is only accessible via RPCs (it's not actually deployed on-chain, and thus can't be called by smart contracts). This reference page documents the specific calls available in the NodeInterface. For a more conceptual description of what it is and how it works, please refer to theNodeInterface conceptual page .

NodeInterface methods

Method Solidity interface Go implementation Description estimateRetryableTicket(address sender, uint256 deposit, address to, uint256 l2CallValue, address excessFeeRefundAddress, address callValueRefundAddress, bytes calldata data) Interface Implementation Estimates the gas needed for a retryable submission constructOutboxProof(uint64 size, uint64 leaf) Interface Implementation Constructs an outbox proof of an l2->l1 send's existence in the outbox accumulator findBatchContainingBlock(uint64 blockNum) Interface Implementation Finds the L1 batch containing a requested L2 block, reverting if none does getL1Confirmations(bytes32 blockHash) Interface Implementation Gets the number of L1 confirmations of the sequencer batch producing the requested L2 block gasEstimateComponents(address to, bool contractCreation, bytes calldata data) Interface Implementation Same as native gas estimation, but with additional info on the l1 costs gasEstimateL1Component(address to, bool contractCreation, bytes calldata data) Interface Implementation Estimates a transaction's l1 costs legacyLookupMessageBatchProof(uint256 batchNum, uint64 index) Interface Implementation Returns the proof necessary to redeem a message nitroGenesisBlock()Interface Implementation Returns the first block produced using the Nitro codebase blockL1Num(uint64 l2BlockNum) Interface Implementation Returns the L1 block number of the L2 block l2BlockRangeForL1(uint64 blockNum) Interface Implementation Finds the L2 block number range that has the given L1 block number Edit this page Last updatedonMar 22, 2024 Previous NodeInterface overview Next Token bridging overview