Frequently asked questions: Run a node

{"@context":"https://schema.org","@type":"FAQPage","mainEntity":[{"@type":"Question","name":"How do I run a node?","acceptedAnswer":{"@type":"Answer","text":"

See instructions here!

\n\n

\n\n"}},{"@type":"Question","name":"How to verify the integrity of the Nitro database I currently have?","acceptedAnswer":{"@type":"Answer","text":"

We have an accumulator hash on all messages, which means that a message can't be added to the database without the previous message being correct.

\n\n

To confirm that everything's working properly, you could just make sure that it's syncing and that the latest block is consistent with other Arbitrum nodes; e.g., you could check it against Arbiscan (note that Arbiscan's search field doesn't support searching by block hash).

\n\n

\n\n"}},{"@type":"Question","name":"How can I check if the node is running properly and diagnose the issue if it is not?","acceptedAnswer":{"@type":"Answer","text":"

We have trace-level logging RPC request implemented on our node. You could use it to log all requests and responses at the trace level. (The performance impact of this should be negligible compared to the network overhead of an RPC request in the first place, especially considering that the request/response will only be serialized for logging if that log level is enabled.)

\n\n

\n\n"}},{"@type":"Question","name":"Why do I need an L1 node to run an Arbitrum node?","acceptedAnswer":{"@type":"Answer","text":"

On the node syncing stage, Arbitrum nodes read transactions from batches that were previously posted on L1 and have been executed. They then connect to the Sequencer feed to receive new incoming batched transactions that have not yet been posted on L1.

\n\n

When fully synced, the Arbitrum node uses the State Transition Function (STF) to consume transactions coming from the Sequencer feed and creates a new state. It also waits for the L1 batch to be posted. If the L1 batch that is finalized on L1 is different from what the Sequencer published, the node will change the state based on the L1 batched transactions.

\n\n

\n\n"}},{"@type":"Question","name":"Can I run an Arbitrum node in p2p mode?","acceptedAnswer":{"@type":"Answer","text":"

Arbitrum doesn't have a consensus mechanism, so \"p2p mode\" doesn't apply. For nodes to sync to the latest chain state, they connect to an L1 node to sync the chain's history that's been posted in calldata and connect to the Sequencer feed for the transactions that have yet to be posted in batches. In no case do nodes need to peer up and sync with each other.

\n\n"}},{"@type":"Question","name":"How do I read messages from the Sequencer feed?","acceptedAnswer":{"@type":"Answer","text":"

Running an Arbitrum relay locally as a Feed Relay lets you subscribe to the Sequencer feed for real-time data as the Sequencer accepts and orders transactions off-chain. Visit How to read the sequencer feed for a detailed how-to.

\n\n

\n\n"}},{"@type":"Question","name":"How do I run a node locally for development?","acceptedAnswer":{"@type":"Answer","text":"

See instructions here.

\n\n

We recommend running nitro nodes via docker; to compile directly / run without docker, you can follow the steps in How to build Nitro locally.

\n\n

\n\n

\n\n

\n\n"}},{"@type":"Question","name":"Is there any way to retrieve pre-Nitro archive data from a Nitro node?","acceptedAnswer":{"@type":"Answer","text":"

The pre-nitro stack is also referred to as the \"classic\" stack. Full nitro nodes start with a database that contains the information from the \"classic\" era.

\n\n

However, it is not possible for a nitro node to query archive information contained in \"classic\" blocks right away. To do that, you need to also run a classic node (instructions here) and set the parameter—node.rpc.classic-redirect=your-classic-node-RPC.

\n\n

Keep in mind that this information only applies to Arbitrum One nodes. Arbitrum Nova and Arbitrum Sepolia nodes started with a Nitro stack from the beginning, so they don't have \"classic\" data.

\n\n

\n\n"}},{"@type":"Question","name":"How can I verify that my node is syncing at a desirable speed?","acceptedAnswer":{"@type":"Answer","text":"

Syncing speed can vary depending on multiple factors. You can find the minimum hardware requirements to run your node in this page. You should also verify your network and disk speed, and make sure that the L1 node is running correctly.

\n\n

\n\n"}},{"@type":"Question","name":"How can I verify that my node is fully synced?","acceptedAnswer":{"@type":"Answer","text":"

You can make an eth_syncing RPC call to your node. When a nitro node is fully synced,eth_syncing returns the value false (just like a normal Geth node).

\n\n

When a nitro node is still syncing, eth_syncing returns a map of values to help understand why the node is not synced. Nitro execution and bottleneck are different from a normal Geth node, so eth_syncing output is unique to nitro.

\n\n

You can find information to understand the output ofeth_syncing in the RPC methods page.

\n\n

\n\n"}},{"@type":"Question","name":"Is there an alternative to Docker when running a node?","acceptedAnswer":{"@type":"Answer","text":"

We recommend running Nitro nodes via Docker, using the guides provided within our documentation. However, you can try to compile the code directly by following the steps described in this guide.

\n\n

\n\n"}},{"@type":"Question","name":"What are the minimum hardware requirements to run a full node?","acceptedAnswer":{"@type":"Answer","text":"

You can see the minimum hardware configuration in this section.

\n\n

\n\n"}},{"@type":"Question","name":"How can I migrate the date of one synced node to a new one?","acceptedAnswer":{"@type":"Answer","text":"

From a fully synced node, you can copy its database (the.arbitrum directory in a default setup) to the same database folder of the new node, and it will start from the same state.

\n\n

Keep in mind that this must be done after a clean shutdown, while the node is not running.

\n\n

\n\n"}},{"@type":"Question","name":"When querying Classic transactions from a Nitro node, I sometimes get incorrect data, like the zero address as the sender. Why is that?","acceptedAnswer":{"@type":"Answer","text":"

Some old Nitro genesis database snapshots didn't properly set the retry sender for Classic blocks and contain said error. If you need to access that information, you can either resync your nitro node with one of the current snapshots, or run a Classic node along with your nitro node and configure a redirection for requests to Classic blocks. Please note that this only happens on Arbitrum One.

\n\n

\n\n"}}]}

How do I run a node?

See instructionshere !

How to verify the integrity of the Nitro database I currently have?

We have an accumulator hash on all messages, which means that a message can't be added to the database without the previous message being correct.

To confirm that everything's working properly, you could just make sure that it'ssyncing and that the latest block is consistent with other Arbitrum nodes; e.g., you could check it againstArbiscan (note that Arbiscan's search field doesn't support searching by block hash).

How can I check if the node is running properly and diagnose the issue if it is not?

We have trace-level logging RPC request implemented on our node. You could use it to log all requests and responses at the trace level. (The performance impact of this should be negligible compared to the network overhead of an RPC request in the first place, especially considering that the request/response will only be serialized for logging if that log level is enabled.)

Why do I need an L1 node to run an Arbitrum node?

On the node syncing stage, Arbitrum nodes read transactions from batches that were previously posted on L1 and have been executed. They then connect to the Sequencer feed to receive new incoming batched transactions that have not yet been posted on L1.

When fully synced, the Arbitrum node uses the State Transition Function (STF) to consume transactions coming from the Sequencer feed and creates a new state. It also waits for the L1 batch to be posted. If the L1 batch that is finalized on L1 is different from what the Sequencer published, the node will change the state based on the L1 batched transactions.

Can I run an Arbitrum node in p2p mode?

Arbitrum doesn't have a consensus mechanism, so "p2p mode" doesn't apply. For nodes to sync to the latest chain state, they connect to an L1 node to sync the chain's history that's been posted in calldata and connect to the Sequencer feed for the transactions that have yet to be posted in batches. In no case do nodes need to peer up and sync with each other.

How do I read messages from the Sequencer feed?

Running an Arbitrum relay locally as aFeed Relay lets you subscribe to the Sequencer feed for real-time data as the Sequencer accepts and orders transactions off-chain. VisitHow to read the sequencer feed for a detailed how-to.

How do I run a node locally for development?

See instructionshere .

We recommend running nitro nodes via docker; to compile directly / run without docker, you can follow the steps inHow to build Nitro locally .

Is there any way to retrieve pre-Nitro archive data from a Nitro node?

The pre-nitro stack is also referred to as the "classic" stack. Full nitro nodes start with a database that contains the information from the "classic" era.

However, it is not possible for a nitro node to query archive information contained in "classic" blocks right away. To do that, you need to also run a classic node (instructions here) and set the parameter—node.rpc.classic-redirect=your-classic-node-

RPC .

Keep in mind that this information only applies to Arbitrum One nodes. Arbitrum Nova and Arbitrum Sepolia nodes started with a Nitro stack from the beginning, so they don't have "classic" data.

How can I verify that my node is syncing at a desirable speed?

Syncing speed can vary depending on multiple factors. You can find the minimum hardware requirements to run your nodein this page . You should also verify your network and disk speed, and make sure that the L1 node is running correctly.

How can I verify that my node is fully synced?

You can make aneth_syncing RPC call to your node. When a nitro node is fully synced,eth_syncing returns the valuefalse (just like a normal Geth node).

When a nitro node is still syncing,eth_syncing returns a map of values to help understand why the node is not synced. Nitro execution and bottleneck are different from a normal Geth node, soeth_syncing output is unique to nitro.

You can find information to understand the output ofeth_syncing in theRPC methods page.

Is there an alternative to Docker when running a node?

We recommend running Nitro nodes via Docker, using the guides provided within our documentation. However, you can try to compile the code directly by following the steps described inthis guide .

What are the minimum hardware requirements to run a full node?

You can see the minimum hardware configurationin this section .

How can I migrate the date of one synced node to a new one?

From a fully synced node, you can copy its database (the.arbitrum directory in a default setup) to the same database folder of the new node, and it will start from the same state.

Keep in mind that this must be done after a clean shutdown, while the node is not running.

When querying Classic transactions from a Nitro node, I sometimes get incorrect data, like the zero address as the sender. Why is that?

Some old Nitro genesis database snapshots didn't properly set the retry sender for Classic blocks and contain said error. If you need to access that information, you can either resync your nitro node with one of thecurrent snapshots , orrun a Classic node along with your nitro node and configure a redirection for requests to Classic blocks. Please note that this only happens on Arbitrum One. Edit this page Last updatedonMar 19, 2024 Previous Troubleshooting: Run a node Next Quickstart: Arbitrum bridge