Sync

warning This article requires a revision.

There are three main synchronization modes

- snap sync* the fastest way to sync to the network (syncs to mainnet in ~3 hours)
-
    - downloads only the latest state, headers, and optionally bodies and receipts
- fast sync* slower than snap sync
-
    - useful on nethermind-only chains (like Gnosis), where snap sync is not available for now
-
    - downloads only the latest state, headers, and optionally bodies and receipts
- archive sync* heavy historical sync verifying all the transactions and keeping all the historical state
-
    - you can run it like this./Nethermind.Runner --config mainnet_archive

Sync Mode Disk Space needed Full current state Full current and all historical states Can sync a full archive node from this Time to sync Time to RPC archive ~12TB YES YES YES ~3 weeks ~3 weeks snap/fast sync with all bodies and receipts ~800GB YES NO YES ~20 hours ~20 hours default snap/fast sync (with barriers set to support Eth2) ~500GB YES NO NO ~11 hours ~11 hours snap/fast sync without receipts ~450GB YES NO YES ~12 hours ~12 hours snap/fast sync without bodies and receipts ~200GB YES NO NO ~9 hours ~9 hours

Snap Sync

Snap sync allows a node to perform the initial synchronization and download of Ethereum’s state up to 10 times faster than using fast sync.

How to Enable

Snap sync is enabled by default for majority of networks. To make sure, check ifSnapSync is set totrue in the Sync module of your.cfg file

"Sync": { "SnapSync": true } IMPORTANT : Do not enable snap sync on a previously synced node. Only use when syncing to the network for the first time.

IMPORTANT : Do not remove other configurations from Sync module. Just add"SnapSync": true , e.g.

"Sync": { "SnapSync": true, "PivotNumber": 15011000, "PivotHash": "0x46c838d02d5fa5bc070080ed7965da1d888f6eb1797045365407c7011280af56", "PivotTotalDifficulty": "52367203434576253689712", "FastBlocks": true }

Snap Sync vs Other Sync Modes

More than 12TB of storage is needed today to run a full archive node — one that stores all the state since genesis. Because of that, setting up an archive node can take days or even weeks. Fast Sync can still take more than 24 hours on the fast machine and download about 90GB state data. With Snap Sync, sync time is reduced to 2-3h with a download of about 30GB.

This reduction in sync time and download size has to do with the specific way in which Ethereum’s state is stored in a node: Merkle trees.

With Fast Sync, a node downloads the headers of each block and retrieves all the nodes beneath it until it reaches the leaves. By contrast, Snap Sync only downloads the leaf nodes, generating the remaining nodes locally which saves time and packets downloaded.

Current limitations and future development

For now Snap Sync on the Nethermind client can only download the Ethereum state but not serve it to other clients - snap serving development is in progress, expected late 2023/early 2024.

The only Ethereum client that supports serving Snap Sync requests is Geth, so only networks supported by Geth can be synced using that method: Mainnet, Goerli, Sepolia.

Fast Sync

After completing the fast sync your node will have the ability to answer questions like'what is my account balancenow ' ,'how many XYZ tokens SomeExchange holdsat the moment ' .

Fast sync has multiple stages. Nethermind uses apivot block number to improve fast sync performance. Thepivot block data is automatically updated after initialization of the client and consists of theblock number ,block hash andblock total difficulty (please note thattotal difficulty is different thandifficulty). Before synchronizing state data Nethermind synchronizes in two directions - backwards frompivot block to 0 for headers and forward to the head of the chain for headers, blocks and receipts. Forward sync might be very slow (5 - 50 bps) so having fresh pivot block is very important - it is guaranteed by recently implemented auto-update.

After downloading the block data Nethermind will start state sync (downloading the latest state trie nodes). We are providing an estimate for the download size and progress but the real value may be different than the estimate (especially if you are using an old version of Nethermind as we sometimes manually adjust the estimator based on our observations of the chain growth rate). Because of this sometimes your sync may continue even when it shows ~100% finished. The other important component is the speed of your sync - if your IO / network / file system causes the state sync to go much slower than around 1.5MB per second on average then you will start downloading some parts of the trie over and over again. In such cases you may be surprised to see something like 58000MB / 53000MB (100%). It means that you downloaded around 5GB of data that is no longer needed. If your sync is very slow (extended beyond two days) then very likely your setup cannot catch up with the chain progress.

After the state sync finishes you will see the'Processed...' messages like in archive sync - it means that your node is in sync and is processing the latest blocks.

Mainnet sync, at the time of writing (December 2020), takes around 8 hours on an UpCloud 16GB RAM 8 CPU 40 VM (and then syncs receipts and bodies in the background if you enabled them in the configuration). Goerli sync should take around 40 minutes.

State sync log messages have multiple values displayed. Firstdd.HH:mm:ss total state sync time is displayed, followed by an estimated sync progress (percentage of total database data synced), then the current download speed is displayed (there will be times when it will slow down to 0 or single digit numbers, especially towards the end of the sync). In general 6 hours sync times shown on screenshots are with around 2000 kB/s (kilobytes per second) average sync rate. You can calculate it in the example as ~45GB / (2MB/s) ~ 22500 seconds ~6.25 hours. We also display the number of state accounts synced, number of trie nodes synced and a diagnostic message in the format of[number_of_pending_requests] .[average time spent in response handler displayed as milliseconds] . So5.6.20ms means that we are awaiting for responses to 5 requests that have been sent to peers and the average time it takes us to process a single response is 6.20ms. The response handling times will differ depending on how many trie nodes are already cached (so they will be significantly slower for a while after the node restart when cache has to be rebuilt) and based on how fast the database IO is (SSD vs NVMe vs cloud drives). For a reasonable sync time you probably should expect these values to be below 15ms (but they may be as high as 700ms for a while after restarting the node).

A single restart of the node during the fast sync may extend the sync time by up to two hours because the node has to rebuild the caches by reading millions of values from the database.

At the last stages of the sync the node will be repeatedly displaying the branch sync progress and changing the block number to which it tries to catch up. This stage should take between 30 minutes and two hours. If it lasts much more then it is possible that you will not be able to catch up with the network progress.

One of the best indicators that you are close to be synced is combined ~100% state size progress and nearly 100% branch sync progress.

Archive Sync

Archive sync is the 'heaviest' and slowest sync mode but allows to ask questions like'what was the balance of my account 2 years ago?' ,'how many XYZ token were held in SomeExchange custody in 2017?' .

We have prepared default archive sync configurations and they can be launched from Nethermind Launcher (just choose the archive options) or by simply loading appropriate config when launching./Nethermind.Runner --config mainnet_archive

./Nethermind.Runner --config goerli_archive

While for some smaller networks archive sync can complete very quickly (in minutes or hours) mainnet sync would take 2 - 6 weeks depending on the speed of your IO (whether you use SSD or NVMe or depending on the cloud provider IOPS). Database size in archive sync is the biggest from all modes as you will store all the historical data.

Explanation of some data in the logs:

- at the beginning you may see a'Waiting for peers...'
- message while the node is trying to discover nodes that it can
- sync with.
- 'Downloaded 1234/8000000'

- shows the number of unprocessed blocks (with transactions) downloaded from the network.
- Formainnet
- this value may be slower than processing at first but very quickly you will see blocks being downloaded
- much faster than processed. Empty blocks can be as small as 512 bytes (just headers without transactions) and full
- blocks with heavy transactions can reach a few hundred kilobytes. We display both current download speed (calculated
- in the last second) and average (total) speed since starting the node.
- 'Processed ...'
- displays the blocks that have been processed by the EVM. The first number shows the current head
- block number, then you can seemgasps
- (million gas per second) - current and total, thentps
- (transactions per
- second) - current and total,bps
- (blocks per second). Thenrecv queue
- (transactions signature public key recovery
- queue),proc queue
- (processor queue). Both recovery queue and processor queue are designed so when too many blocks
- are waiting for processing then only their hashes are kept in memory and remaining data are stored in the database.
- Thus, the queues numbers that you can see will be capped by some number.
- 'Cache for epoch...'
- informs aboutethash
- cache needed for block seal verification (only onmainnet
- andropsten
-). Caches will be calculated every 30000 blocks (length of an epoch) but can also be calculated for the
- latest blocks that are being broadcast on the network.
- After the archive sync finishes you will see the'Processed...'
- message appearing on average every 15 seconds when
- the new block is processed.
- mgasps
- ,tps
- ,bps
- values should not be treated as comparable as they may differ massively on different parts of
- the chain. For example when blocks are empty you may see very highbps
- values with very low (or even zero)tps
- andmgasps
- values as there are no transactions and no gas for EVM processing and blocks are very light. On the other
- hand when blocks are filled with very heavy transactions thenbps
- might be very low whilemgasps
- will be very
- high. It is even possible that you will see a lot of very light transactions wheretps
- will be high whilebps
- andmgasps
- will be average.

Sync time

Sync time heavily depends on the hardware used for the node, network speed, and peering. We are constantly pursuing to make it as fast as possible. Below is a brief on how the sync time looks on different machines and various chains (tested with Nethermind v1.21.0).

- High-end VM
- Mid-end VM
- Old-spec VM

Hardware configuration:

- Cloud provider: Akamai (formerly Linode)
- CPU: AMD EPYC 7601, 16 vCPU
- Memory: 64 GB

- Storage: 1.2 TB, ~40k IOPS

- Mainnet

- Goerli
- Sepolia
- Gnosis
- Chiado

The high-level data on major sync milestones:

- Attestation time:
- 2h 3m
- Full sync time:
- 7h 3m

The detailed breakdown of sync stages:

- Snap sync phase 1:
- 1h 58m
- State sync:
- 4m
- Old headers:
- 1h 27m
- Old bodies:
- 1h 55m
- Old receipts:

- 3h 2m The high-level data on major sync milestones:

- Attestation time:

- 2h 49m
- Full sync time:
- 4h 58m

The detailed breakdown of sync stages:

- Snap sync phase 1:
- 2h 49m
- State sync:
- 0.5m
- Old headers:
- 11m
- Old bodies:
- 1h 2m
- Old receipts:

- 1h 5m The high-level data on major sync milestones:

- Attestation time:

- 8m
- Full sync time:
- 1h 9m

The detailed breakdown of sync stages:

- Snap sync phase 1:
- 8m
- State sync:
- 0.3m
- Old headers:
- 12m
- Old bodies:
- 21m
- Old receipts:

- 22m The high-level data on major sync milestones:

- Attestation time:

- 13h 40m
- Full sync time:
- 17h 17m

The detailed breakdown of sync stages:

- State sync:
- 13h 40m

- Old headers:
- 1h 46m
- Old bodies:
- 1h 31m
- Old receipts:

- 2h 3m The high-level data on major sync milestones:

- Attestation time:

- 20m
- Full sync time:
- 40m

The detailed breakdown of sync stages:

- State sync:
- 20m
- Old headers:
- 11m
- Old bodies:
- 8m
- Old receipts:

- 10m Hardware configuration:

- Cloud provider: AWS

- c7g.2xlarge
- : 8 vCPU, 16 GiB memory

- Storage: 1 TB, ~10k IOPS

- Mainnet

- Goerli
- Sepolia
- Gnosis
- Chiado

The high-level data on major sync milestones:

- Attestation time:
- 5h 55m
- Full sync time:
- 12h 37m

The detailed breakdown of sync stages:

- Snap sync phase 1:
- 4h 35m
- State sync:
- 1h 20m
- Old headers:
- 1h 43m
- Old bodies:
- 2h 13m
- Old receipts:

- 4h 28m The high-level data on major sync milestones:

- Attestation time:

- 1h 32m
- Full sync time:
- 4h 10m

The detailed breakdown of sync stages:

- Snap sync phase 1:
- 1h 19m
- State sync:

- 12m
- Old headers:
- 23m
- Old bodies:
- 49m
- Old receipts:

- 1h 35m The high-level data on major sync milestones:

- Attestation time:

- 17m
- Full sync time:
- 1h 3m

The detailed breakdown of sync stages:

- Snap sync phase 1:
- 13m
- State sync:
- 4m
- Old headers:
- 15m
- Old bodies:
- 19m
- Old receipts:

- 29m The high-level data on major sync milestones:

- Attestation time:

- 15h 54m
- Full sync time:
- 18h 28m

The detailed breakdown of sync stages:

- State sync:
- 15h 54m
- Old headers:
- 1h 4m
- Old bodies:
- 40m
- Old receipts:

- 1h 52m The high-level data on major sync milestones:

- Attestation time:

- 13m
- Full sync time:
- 25m

The detailed breakdown of sync stages:

- State sync:
- 13m
- Old headers:
- 12m
- Old bodies:
- 5m
- Old receipts:

- 5m Hardware configuration:

- Cloud provider: Scaleway

- CPU: Intel Xeon Processor E5-2620 v2, 2 vCPU
- Memory: 192 GB

- Storage: 1 TB, ~44k IOPS

- Mainnet

- Goerli
- Sepolia
- Gnosis
- Chiado
- Energy Web
- Volta

The high-level data on major sync milestones:

- Attestation time:
- 5h 55m
- Full sync time:
- 17h 1m

The detailed breakdown of sync stages:

- Snap sync phase 1:
- 4h 29m
- State sync:
- 25m
- Old headers:
- 1h 27m
- Old bodies:
- 3h 39m
- Old receipts:

- 8h 3m The high-level data on major sync milestones:

- Attestation time:

- 1h 51m
- Full sync time:
- 5h 55m

The detailed breakdown of sync stages:

- Snap sync phase 1:
- 1h 40m
- State sync:
- 11m
- Old headers:
- 50m
- Old bodies:
- 1h 34m
- Old receipts:

- 2h 14m The high-level data on major sync milestones:

- Attestation time:

- 16m
- Full sync time:
- 2h 9m

The detailed breakdown of sync stages:

- Snap sync phase 1:
- 15m
- State sync:
- 1m
- Old headers:
- 26m
- Old bodies:
- 45m
- Old receipts:

- 56m The high-level data on major sync milestones:

- Attestation time:

- 15h 13m
- Full sync time:
- 17h 30m

The detailed breakdown of sync stages:

- State sync:
- 15h 13m
- Old headers:
- 3h 8m
- Old bodies:
- 50m
- Old receipts:

- 1h 25m The high-level data on major sync milestones:

- Attestation time:

- 20m
- Full sync time:
- 40m

The detailed breakdown of sync stages:

- State sync:
- 4m
- Old headers:
- 1h 27m
- Old bodies:
- 1h 55m
- Old receipts:

- 3h 2m The detailed breakdown of sync stages:

- State sync:

- 13h 7m
- Old headers:
- 2h 32m
- Old bodies:
- 51m
- Old receipts:
- 1h 11m
- Full sync time:

- 15h 20m The detailed breakdown of sync stages:

- State sync:

- 14h 27m
- Old headers:
- 2h 42m
- Old bodies:
- 40m
- Old receipts:
- 58m
- Full sync time:
- 16h 10m

Resync a node from scratch

This guide explains how to resync a Nethermind node using the existing Pivot block or updating it to a more recent one.

Steps to Resync a Nethermind Node

1. Stop the Nethermind node
2. : If your Nethermind node is currently running, stop it to ensure that no new data is
3. being written to the database during the resync process.
4. Delete the existing database
5. : Navigate to the Nethermind data directory. The location of this directory depends

6. on how Nethermind was installed and your configuration settings. Inside the data directory, find thenethermind_db
7. folder and delete themainnet
8. subfolder to remove the existing database for the mainnet.
9. Update the configuration file (optional)
10. : If you want to change any configuration settings before resyncing the
11. node, edit themainnet.cfg
12. file located in the Nethermind directory. For example, you might want to adjust the
13. pruning settings or specify a different network.
14. Update the Pivot block (optional)
15. \danger
16. Only for versions before 1.19.0 where Auto-Pivot approach was introduced
17.
    1. Using Etherscan
18.
    1. : If you want to speed up the syncing process, you can update the Pivot block to a more recent
19.
    1. one. To do this, find theSync
20.
    1. section in themainnet.cfg
21.
    1. file and update thePivotNumber
22.
    1. andPivotHash
23.
    1. fields to match a recent "finalized" block number and its corresponding hash. You can obtain this information from
24.
    1. a block explorer such asEtherscan
25.
    1. .
26.
    1. Using block number 17165278 fromEtherscan
27.
    1. :
28.
    1. {
29.
    1. "PivotNumber": 17165278,
30.
    1. "PivotHash": "0xa665315efd923f3b11215feee09a9d3e13c5e6ee602fa19b642824682ec0a752"
31.
    1. }
32.
    1. Using Nethermind's GitHub
33.
    1. : Alternatively, you can update the Pivot block by referring to
34.
    1. theNethermind's mainnet.cfg file on GitHub
35.
    1. .
36.
    1. The Pivot block is periodically bumped to the HEAD-8192 block of the mainnet chain. Copy thePivotNumber
37.
    1. andPivotHash
38.
    1. values from the GitHub file and update your localmainnet.cfg
39.
    1. file accordingly.
40. Restart the Nethermind node
41. : Start the Nethermind node again to initiate the resync process. The node will begin
42. syncing from the existing Pivot block or the specified updated Pivot block, downloading and processing all the blocks
43. in the blockchain.

To ensure that your Nethermind node is resyncing, you can monitor the logs for the node's progress. The logs will display information about the block processing, synchronization status andOldHeaders being processed. By observing the increasing block numbers and synchronization messages in the logs, you can confirm that the resync process is active and working as expected.

Old Headers 0 / 17154000 | queue 0 | current 0.00bps | total 0.00bps

Old Headers 768 / 17154000 | queue 0 | current 766.07bps | total 762.49bps

Beacon Headers from block 17154001 to block 17169722 | 960 / 15722 | queue 4992 | current 0.00bps | total 40622848.83bps

Old Headers 9024 / 17154000 | queue 0 | current 576.40bps | total 1286.40bps

Beacon Headers from block 17154001 to block 17169723 | 9024 / 15723 | queue 6698 | current 2694.81bps | total 3882943.63bps Downloaded 17154031 / 17172359 | current 0.00bps | total 0.00bps

Downloaded 17154062 / 17169724 | current 0.00bps | total 2.88bps Keep in mind that resyncing a Nethermind node can take a considerable amount of time. It depends on your hardware, internet connection, and the size of the blockchain. Edit this page Last updatedonFeb 17, 2024 Previous Configuration Next Security