

EN Observability

On This Page * [Buckets](#) * [Metrics](#) * [Interpretation](#) *]

#

EN Observability

The EN provides several options for setting up observability. Configuring logs and sentry is described in the configuration section, so this section focuses on the exposed metrics.

This section is written with the assumption that you're familiar with [Prometheus](#)[open in new window](#) and [Grafana](#)[open in new window](#).

#

Buckets

By default, latency histograms are distributed in the following buckets (in seconds):

[0.001, 0.005, 0.025, 0.1, 0.25, 1.0, 5.0, 30.0, 120.0]

#

Metrics

EN exposes a lot of metrics, a significant amount of which aren't interesting outside the development flow. This section's purpose is to highlight metrics that may be worth observing in the external setup.

If you are not planning to scrape Prometheus metrics, please unset `EN_PROMETHEUS_PORT` environment variable to prevent memory leaking.

Metric name	Type	Labels	Description
<code>external_node_synced</code>	Gauge	-	1 if synced, 0 otherwise. Matches <code>eth_call</code> behavior
<code>external_node_sync_lag</code>	Gauge	-	How many blocks behind the main node the EN is
<code>external_node_fetcher_requests</code>	Histogram	-	Duration of requests performed by the different fetcher components
<code>external_node_fetcher_cache_requests</code>	Histogram	-	Duration of requests performed by the fetcher cache layer
<code>external_node_fetcher_miniblock</code>	Gauge	status	The number of the last L2 block update fetched from the main node
<code>external_node_fetcher_l1_batch</code>	Gauge	status	The number of the last batch update fetched from the main node
<code>external_node_action_queue_action_queue_size</code>	Gauge	-	Amount of fetched items waiting to be processed
<code>server_miniblock_number</code>	Gauge	stage = sealed	Last locally applied L2 block number
<code>server_block_number</code>	Gauge	stage = sealed	Last locally applied L1 batch number
<code>server_block_number</code>	Gauge	stage = tree_lightweight_mode	Last L1 batch number processed by the tree
<code>server_processed_txs</code>	Counter	-	TPS values
<code>api_web3_call</code>	Histogram	-	Duration of Web3 API calls
<code>sql_connection_acquire</code>	Histogram	-	Time to get an SQL connection from the connection pool

#

Interpretation

After applying a dump, the EN has to rebuild the Merkle tree to verify the correctness of the state in PostgreSQL. During this stage, `server_block_number { stage='tree_lightweight_mode' }` is increasing from 0 to `server_block_number { stage='sealed' }`, while the latter does not increase (EN needs the tree to be up-to-date to progress).

After that, the EN has to sync with the main node. `server_block_number { stage='sealed' }` is increasing, and `external_node_sync_lag` is decreasing.

Once the node is synchronized, it is indicated by the `external_node_synced`.

Metrics can be used to detect anomalies in configuration, which is described in more detail in the next section.

[\[\] Edit this page](#)[open in new window](#) Last update: Contributors: [\[\[albi\]codes\]](#)]

[Prev API Overview](#) [Next Troubleshooting](#)