# 10.1.4 against 10.1.1

value-only workload

Michael Karg, Cardano Performance team

2025-01-08

# Contents

# Chapter 1

# Manifest

We compare `10.1.4` (Conway) relative to `10.1.1` (Conway), under value-only workload.

| | 10.1.1 | 10.1.4 |
|---|---|---|
| Analysis date | 2024-10-20 | 2025-01-08 |
| Cluster system start date | 2024-10-19 | 2025-01-07 |
| Cluster system start time | 08:35:58 | 11:55:06 |
| Identifier | 10.0 | 10.1.4 |
| Run batch | 10.0.0-pre | 10.1.4 |
| GHC version | 8.10.7 | 8.10.7 |
| cardano-node version | 10.0 | 10.1.4 |
| ouroboros-consensus version | 0.21.0.0 | 0.21.0.0 |
| ouroboros-network version | 0.17.1.2 | 0.17.1.2 |
| cardano-ledger-core version | 1.15.0.0 | 1.15.0.0 |
| plutus-core version | 1.36.0.0 | 1.36.0.0 |
| cardano-crypto version | 1.1.2 | 1.1.2 |
| cardano-prelude version | 0.2.0.0 | 0.2.0.0 |
| cardano-node git | cdb45dd | b2b3cfb |
| ouroboros-consensus git | 358305b | 358305b |
| ouroboros-network git | df3431f | df3431f |
| cardano-ledger-core git | 0ba8e73 | 0ba8e73 |
| plutus-core git | fb82893 | fb82893 |
| cardano-crypto git | 6568a5e | 6568a5e |
| cardano-prelude git | 273167c | 273167c |
| Era | conway | conway |
| Delegation map size | 1000000 | 1000000 |
| Starting UTxO set size | 4000000 | 4000000 |
| DRep count | 10000 | 10000 |
| Extra tx payload | 100 | 100 |
| Tx inputs | 2 | 2 |
| Tx Outputs | 2 | 2 |
| TPS | 12.0 | 12.0 |
| Transaction count | 768000 | 768000 |
| Plutus script | — | — |
| Machines | 52 | 52 |
| Number of filters applied | 3 | 3 |
| Log text lines emitted per host | 5207297.4230 | 5076718.9615 |
| Log objects emitted per host | 5207267.4230 | 5076688.9615 |
| Log objects analysed per host | 2264591.6923 | 2215079.4423 |
| Host run time, s | 63851.9 | 63870.8 |
| Host log line rate, Hz | 81.553 | 79.484 |
| Total log objects analysed | 117758768 | 115184131 |
| Run time, s | 63858 | 63876 |
| Analysed run duration, s | 48014 | 48039 |
| Run time efficiency | 0.75 | 0.75 |
| Node start spread, s | 19.322312 | 6.4738540 |
| Node stop spread, s | 3.2642189 | 3.5671951 |
| Perf analysis start spread, s | 0 | 0 |
| Perf analysis stop spread, s | 3 | 3 |
| Slots analysed | 48012 | 48037 |
| Blocks analysed | 2263 | 2186 |
| Blocks rejected | 878 | 847 |

# Chapter 2

# Analysis

## 2.1 Resource Usage

| | 10.1.1 | 10.1.4 | Δ | Δ% |
|---|---|---|---|---|
| Forge loop starts, # | 0.99827 | 0.99835 | 0.000 | 0 |
| Process CPU usage, % | 8.7403 | 8.7556 | 0.015 | 0 |
| RTS GC CPU usage, % | 1.0667 | 1.0777 | 0.011 | 1 |
| RTS Mutator CPU usage, % | 7.6697 | 7.6668 | -0.003 | 0 |
| Major GCs, # | 0.001 | 0.00104 | 0.000 | 0 |
| Minor GCs, # | 2.0745 | 2.0661 | -0.008 | 0 |
| Kernel RSS, MB | 9137.0 | 9268.7 | 131.700 | 1 |
| RTS heap size, MB | 9084.8 | 9216.6 | 131.800 | 1 |
| RTS live GC dateset, MB | 4000.2 | 4006.7 | 6.500 | 0 |
| RTS alloc rate, MB/s | 64.564 | 64.287 | -0.277 | 0 |
| Filesystem reads, KB/s | 0.003 | 0.00021 | -0.003 | -100 |
| Filesystem writes, KB/s | 240.4 | 255.77 | 15.370 | 6 |
| CPU 85% spans, slots | 3.7809 | 4.0053 | 0.224 | 6 |
| Sample count | (249>) | (249>) | | |

## 2.2 Anomaly control

| | 10.1.1 | 10.1.4 | Δ | Δ% |
|---|---|---|---|---|
| Blocks per host, blocks | 62.615 | 60.538 | -2.077 | -3 |
| Filtered to chained block ratio, / | 0.72055 | 0.72131 | 0.001 | 0 |
| Chained to forged block ratio, / | 0.96428 | 0.96371 | -0.001 | 0 |
| Height & slot battles, blocks | 0.00751 | 0.00365 | -0.004 | -53 |
| Block size, B | 88963 | 88959 | -4 | 0 |
| Sample count | (52) | (52) | | |

## 2.3 Forging

| | 10.1.1 | 10.1.4 | Δ | Δ% |
|---|---|---|---|---|
| Started forge loop iteration, s | 0.00081 | 0.00079 | -0.000 | 0 |
| Acquired block context, s | 0.02697 | 0.02509 | -0.002 | -7 |
| Acquired ledger state, s | 6e-05 | 6e-05 | 0.000 | 0 |
| Acquired ledger view, s | 3e-05 | 2e-05 | -0.000 | 0 |
| Leadership check duration, s | 0.00042 | 0.00043 | 0.000 | 0 |
| Ledger ticking, s | 0.02634 | 0.02785 | 0.002 | 8 |
| Mempool snapshotting, s | 0.07498 | 0.0746 | -0.000 | 0 |
| Leadership to forged, s | 0.00087 | 0.00087 | 0.000 | 0 |
| Forged to announced, s | 0.00076 | 0.00073 | -0.000 | 0 |
| Forged to sending, s | 0.00657 | 0.00759 | 0.001 | 15 |
| Forged to self-adopted, s | 0.08334 | 0.08546 | 0.002 | 2 |
| Slot start to announced, s | 0.13128 | 0.13048 | -0.001 | -1 |
| Sample count | (2263) | (2186) | | |

## 2.4 Individual peer propagation

| | 10.1.1 | 10.1.4 | Δ | Δ% |
|---|---|---|---|---|
| First peer notice, s | 0.13354 | 0.1326 | -0.001 | -1 |
| First peer fetch, s | 0.14414 | 0.14433 | 0.000 | 0 |
| Notice to fetch request, s | 0.00141 | 0.00146 | 0.000 | 0 |
| Fetch duration, s | 0.35543 | 0.35826 | 0.003 | 1 |
| Fetched to announced, s | -0.0 | -0.0 | 0.000 | nan |
| Fetched to sending, s | 0.04572 | 0.04552 | -0.000 | 0 |
| Fetched to adopted, s | 0.08436 | 0.08461 | 0.000 | 0 |
| Sample count | (2263) | (2186) | | |

## 2.5 End-to-end propagation

| | 10.1.1 | 10.1.4 | Δ | Δ% |
|---|---|---|---|---|
| 0.50 adoption, s | 0.68388 | 0.68167 | -0.002 | 0 |
| 0.80 adoption, s | 1.0432 | 1.0602 | 0.017 | 2 |
| 0.90 adoption, s | 1.0629 | 1.0807 | 0.018 | 2 |
| 0.92 adoption, s | 1.0686 | 1.0868 | 0.018 | 2 |
| 0.94 adoption, s | 1.0752 | 1.0931 | 0.018 | 2 |
| 0.96 adoption, s | 1.0837 | 1.1011 | 0.017 | 2 |
| 0.98 adoption, s | 1.097 | 1.1151 | 0.018 | 2 |
| 1.00 adoption, s | 1.1449 | 1.1546 | 0.010 | 1 |
| Sample count | (2263) | (2186) | | |

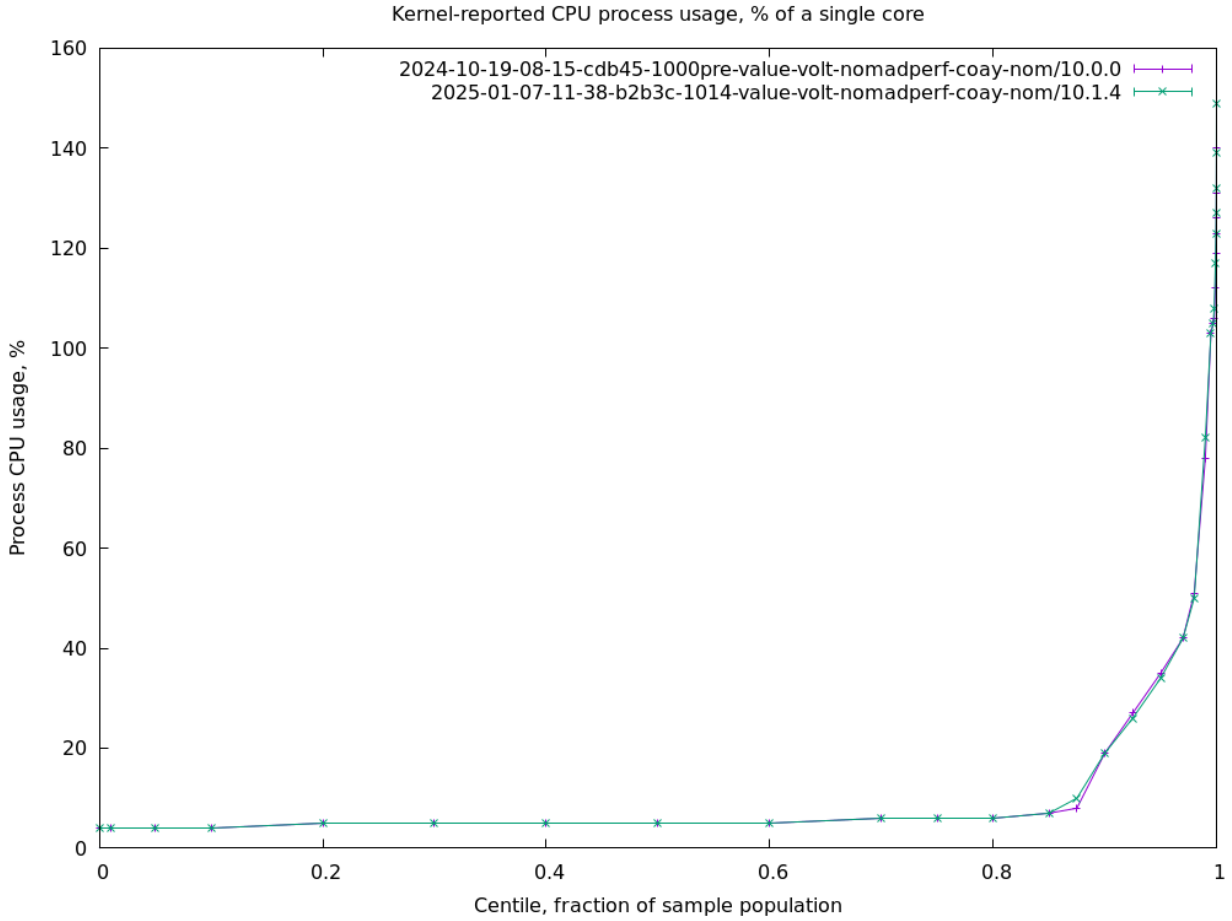
# Part I

# Appendix A: charts

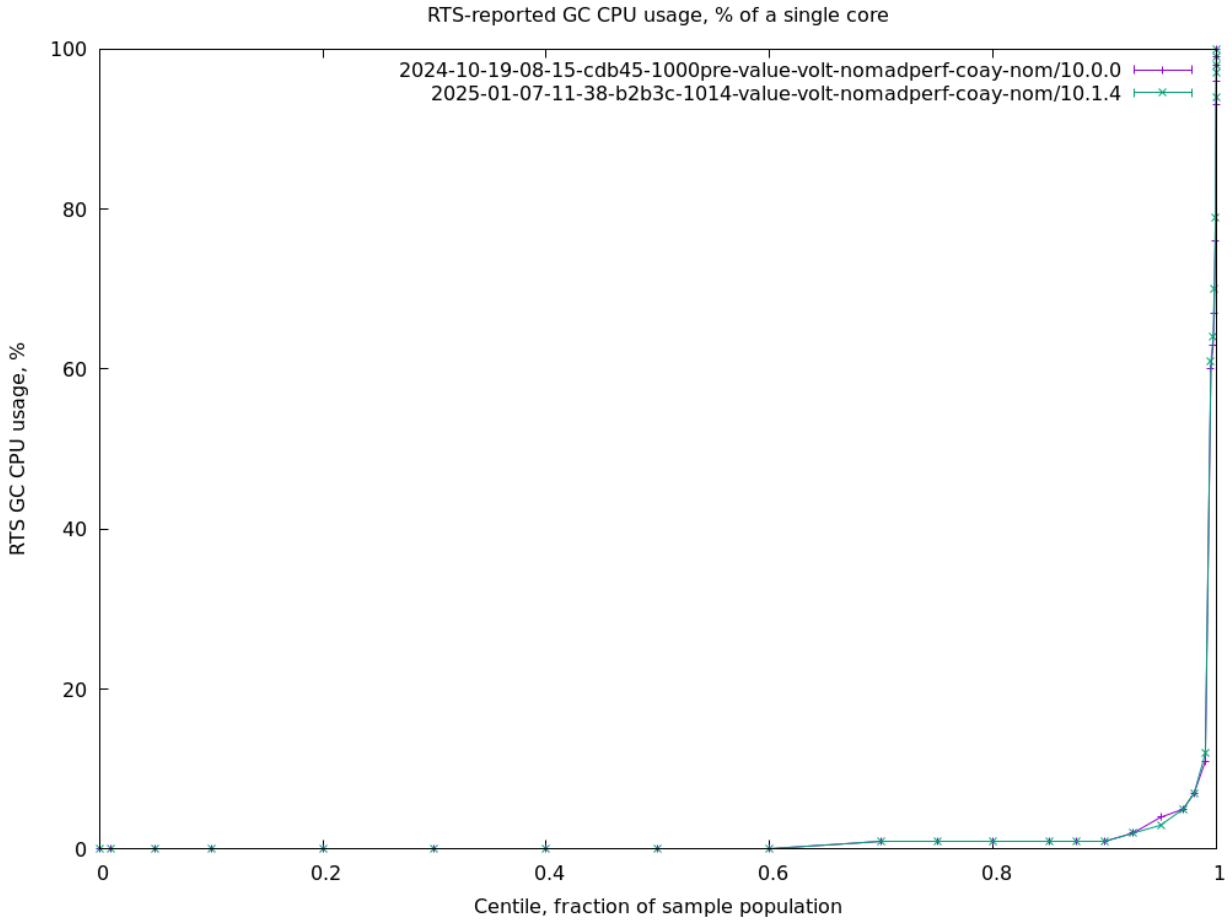
# Chapter 3

# Cluster performance charts

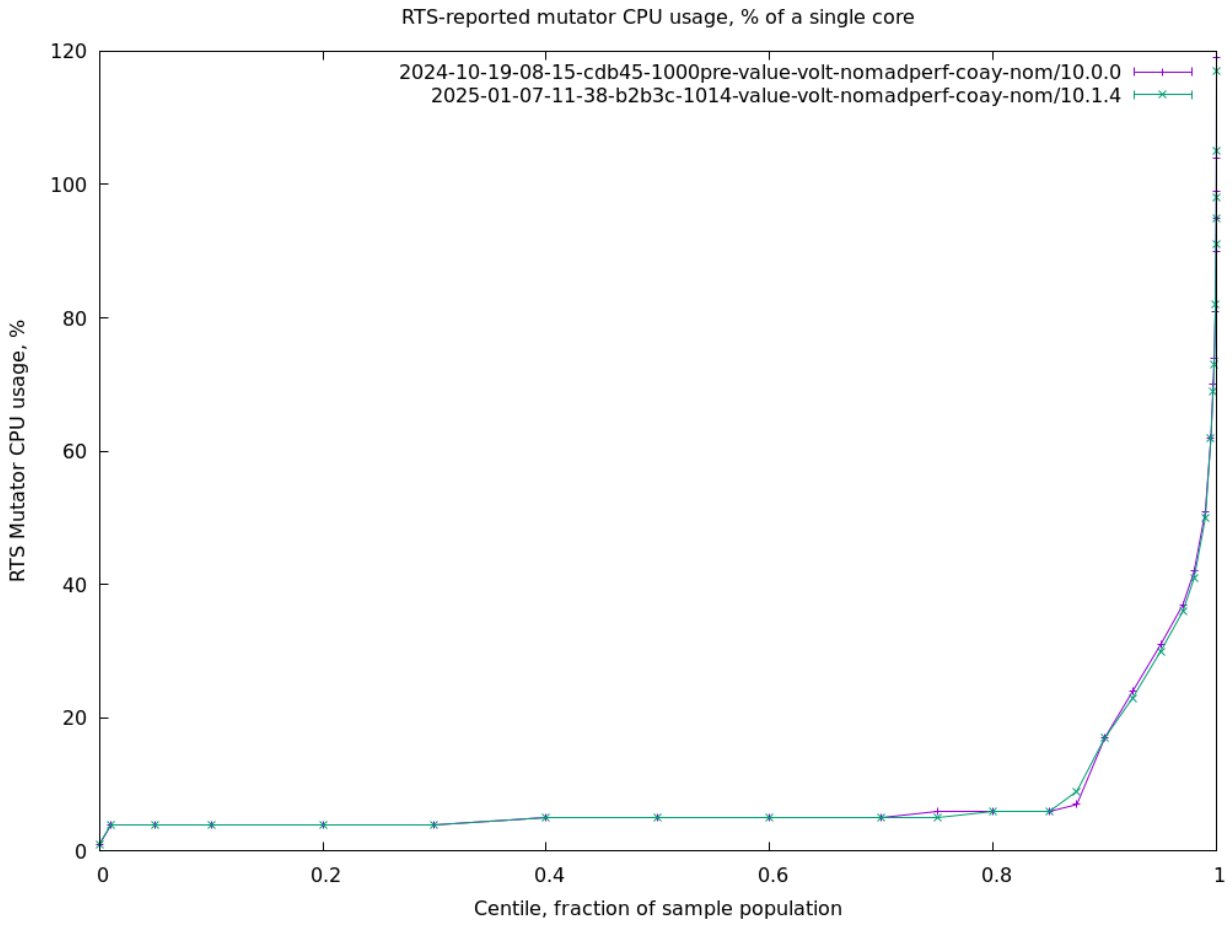
**Process CPU usage (CentiCpu)** Kernel-reported CPU process usage, % of a single core



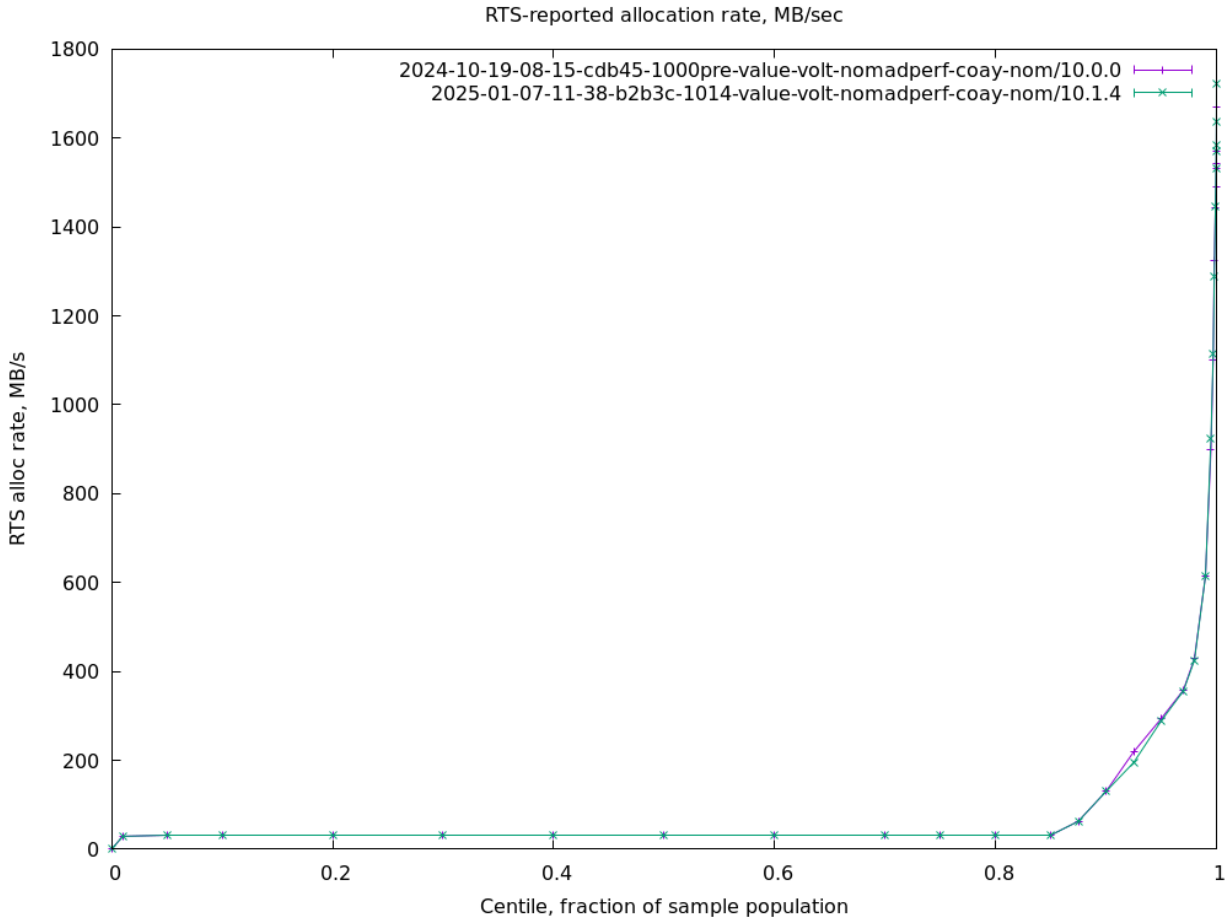

**RTS GC CPU usage (CentiGC)** RTS-reported GC CPU usage, % of a single core



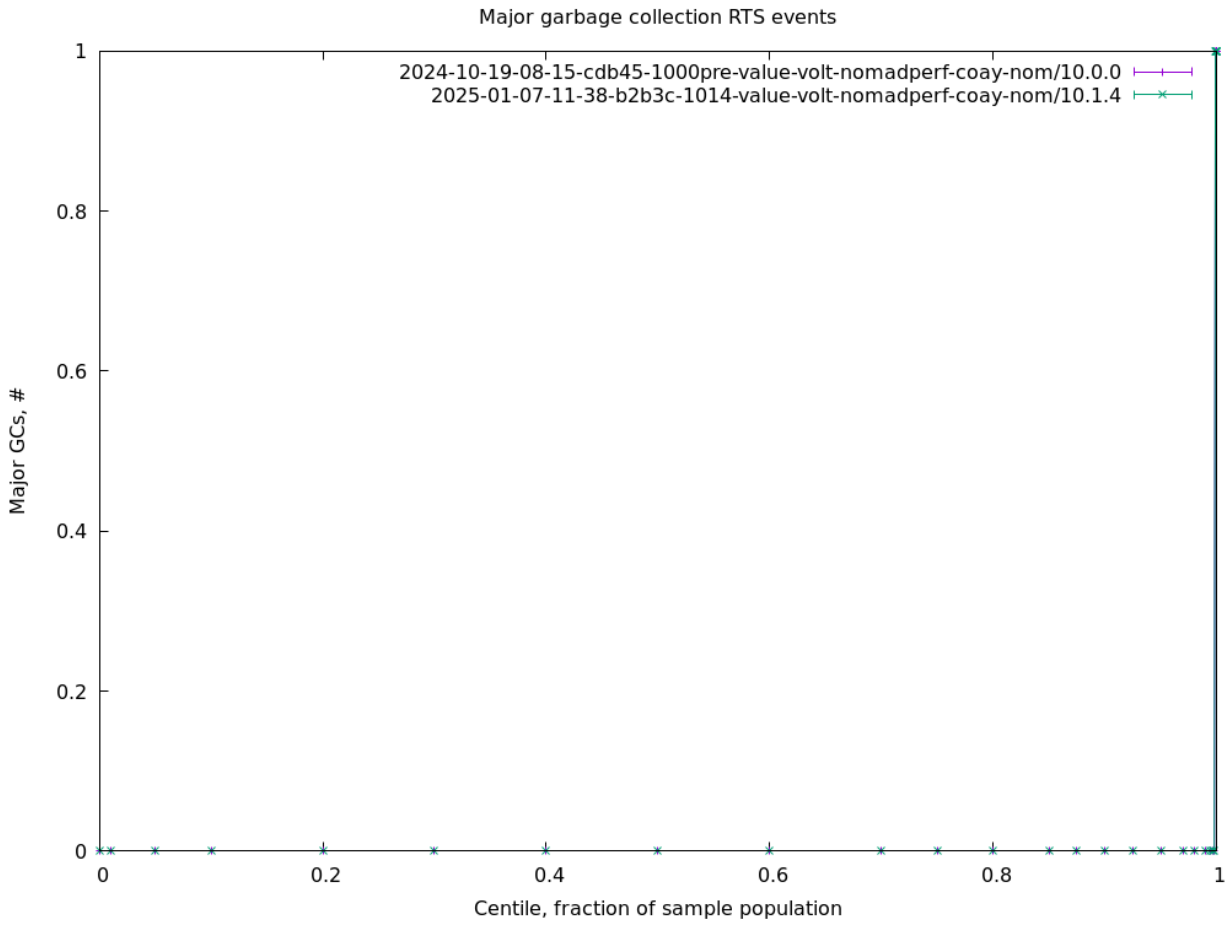

**RTS Mutator CPU usage (CentiMut)** RTS-reported mutator CPU usage, % of a single core



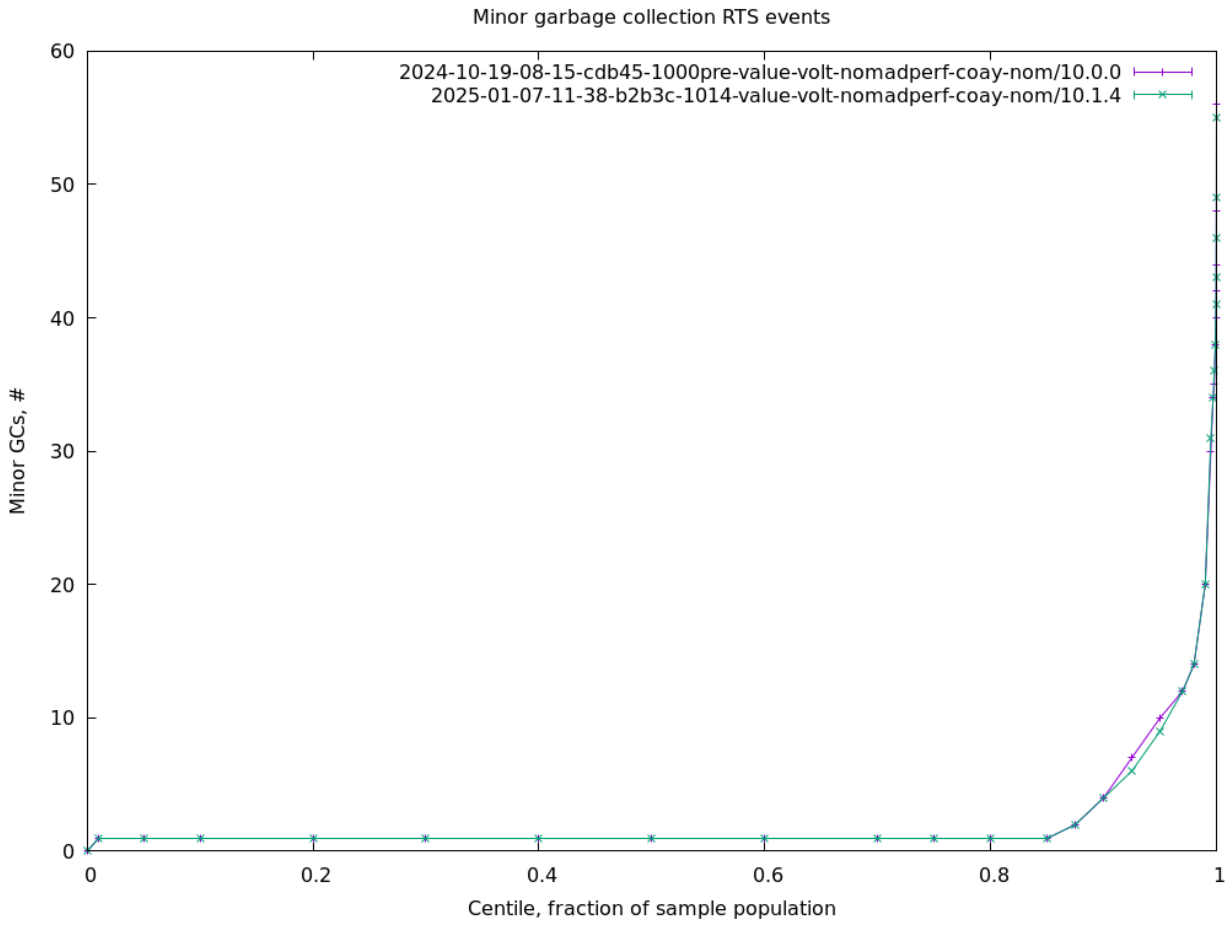

**RTS alloc rate (Alloc)** RTS-reported allocation rate, MB/sec



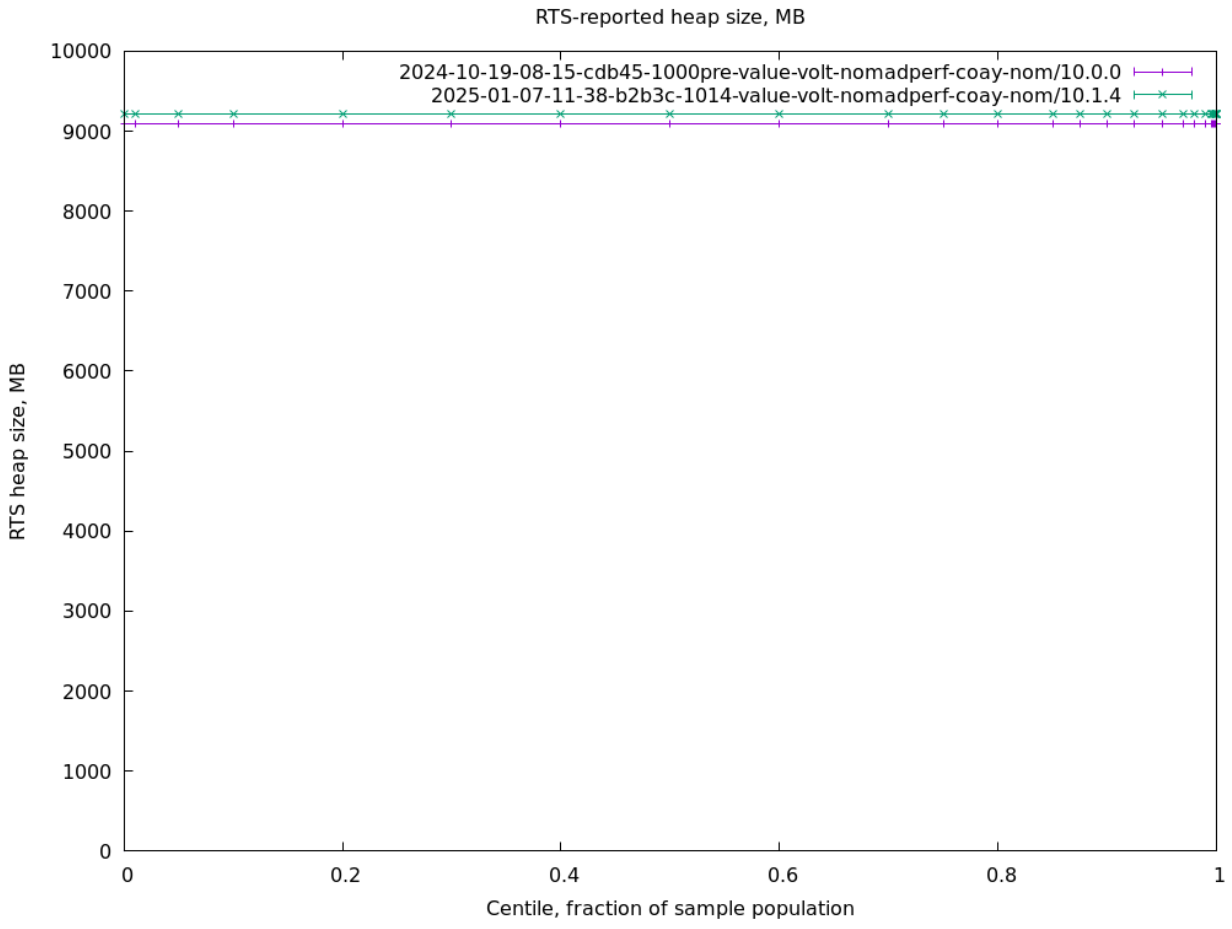

**Major GCs (GcsMajor)** Major garbage collection RTS events



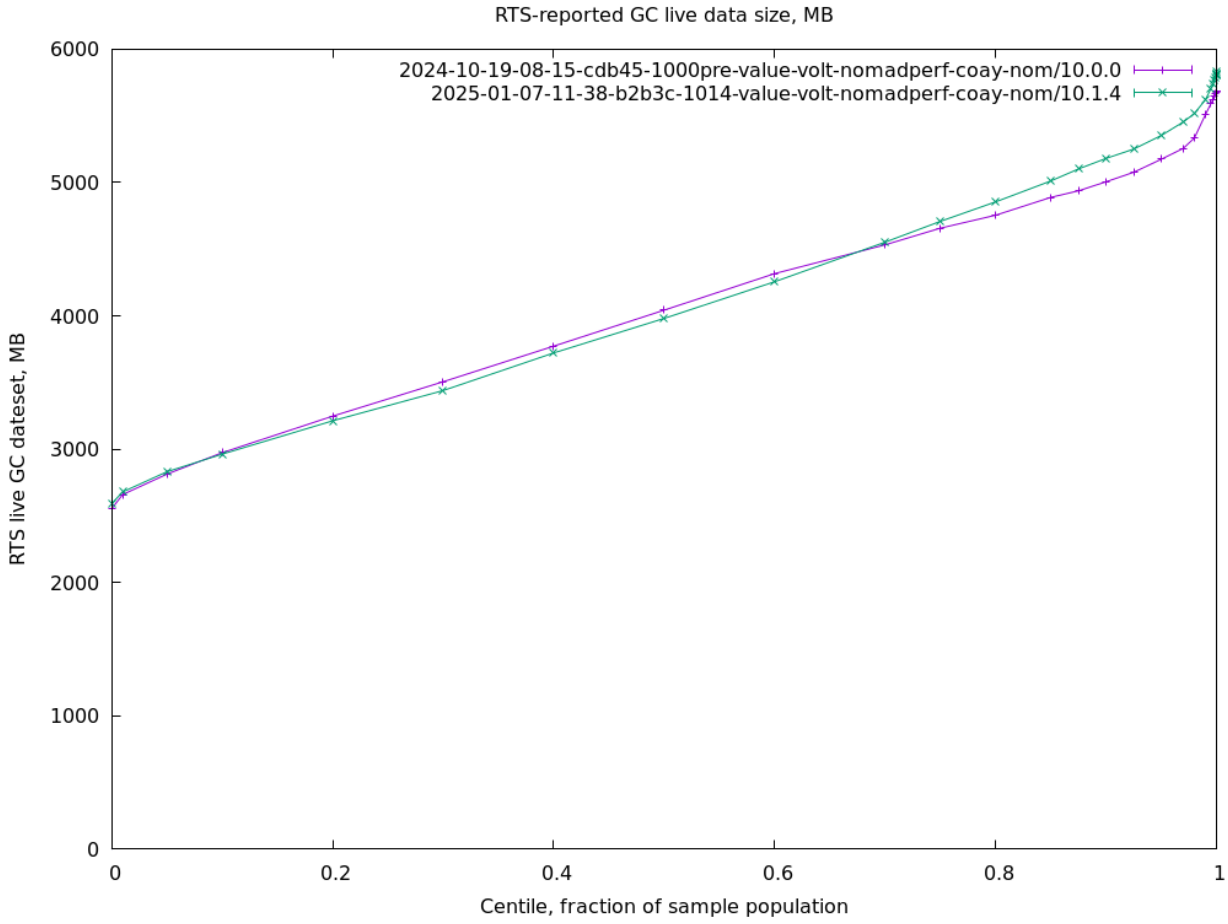

**Minor GCs (GcsMinor)** Minor garbage collection RTS events



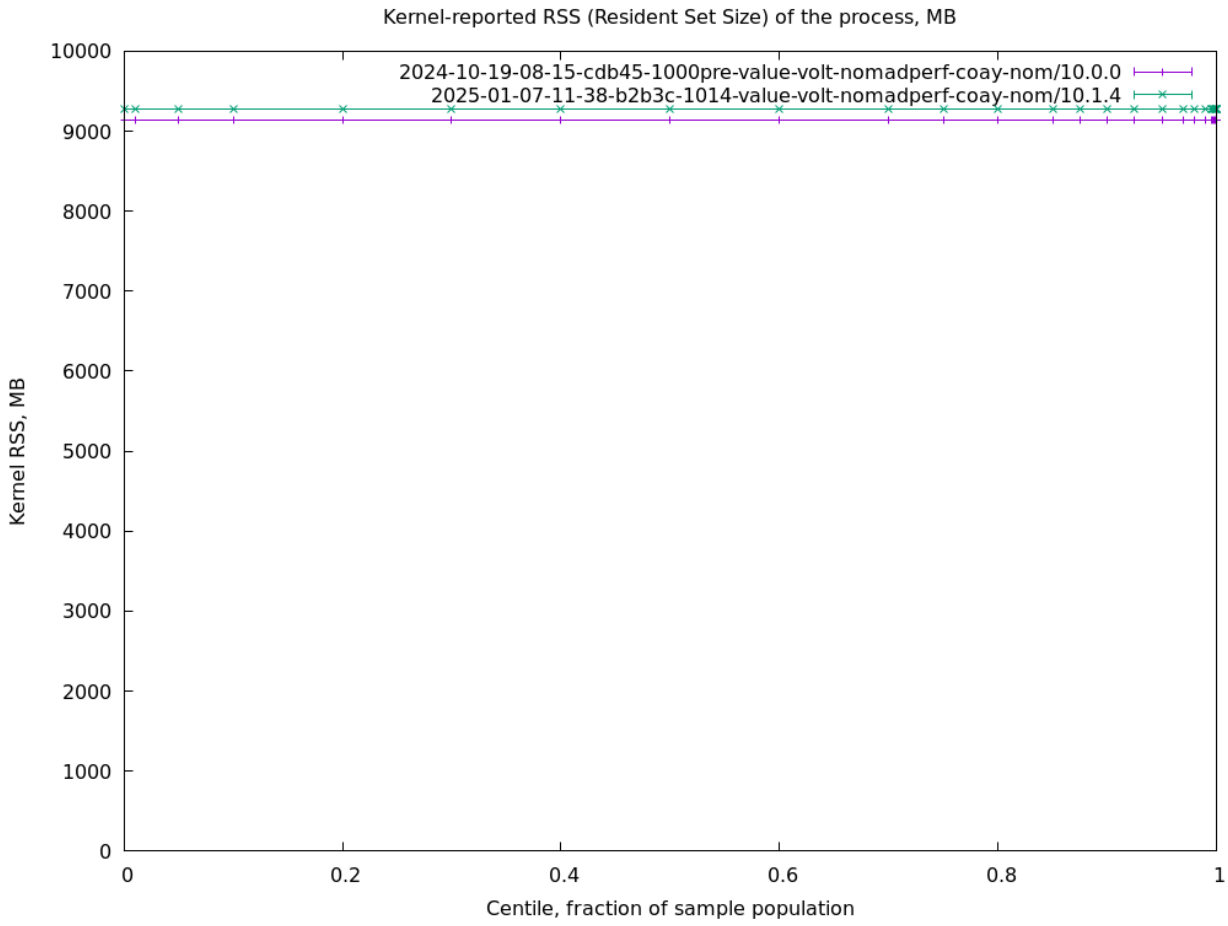

**RTS heap size (Heap)** RTS-reported heap size, MB



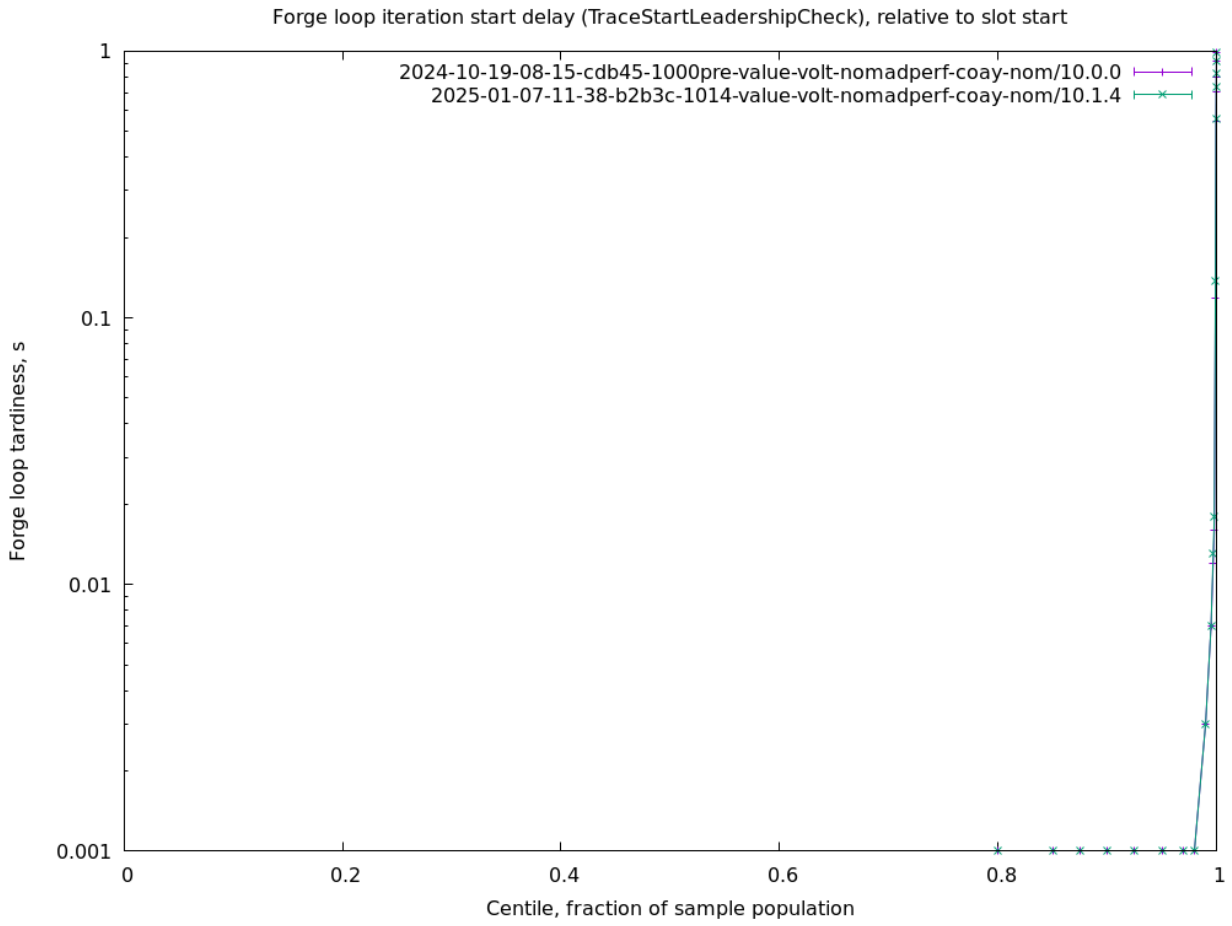

**RTS live GC dateset (Live)** RTS-reported GC live data size, MB



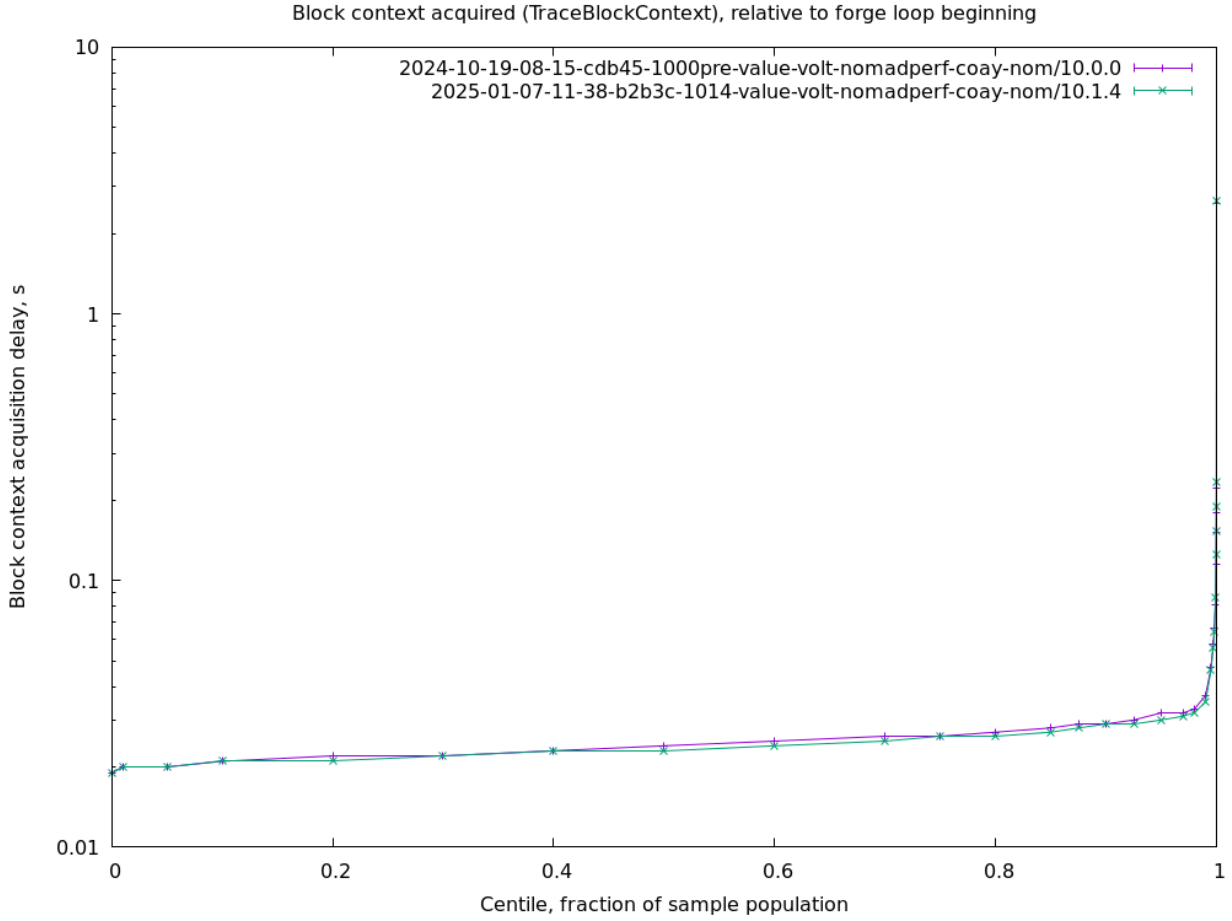

**Kernel RSS (RSS)** Kernel-reported RSS (Resident Set Size) of the process, MB



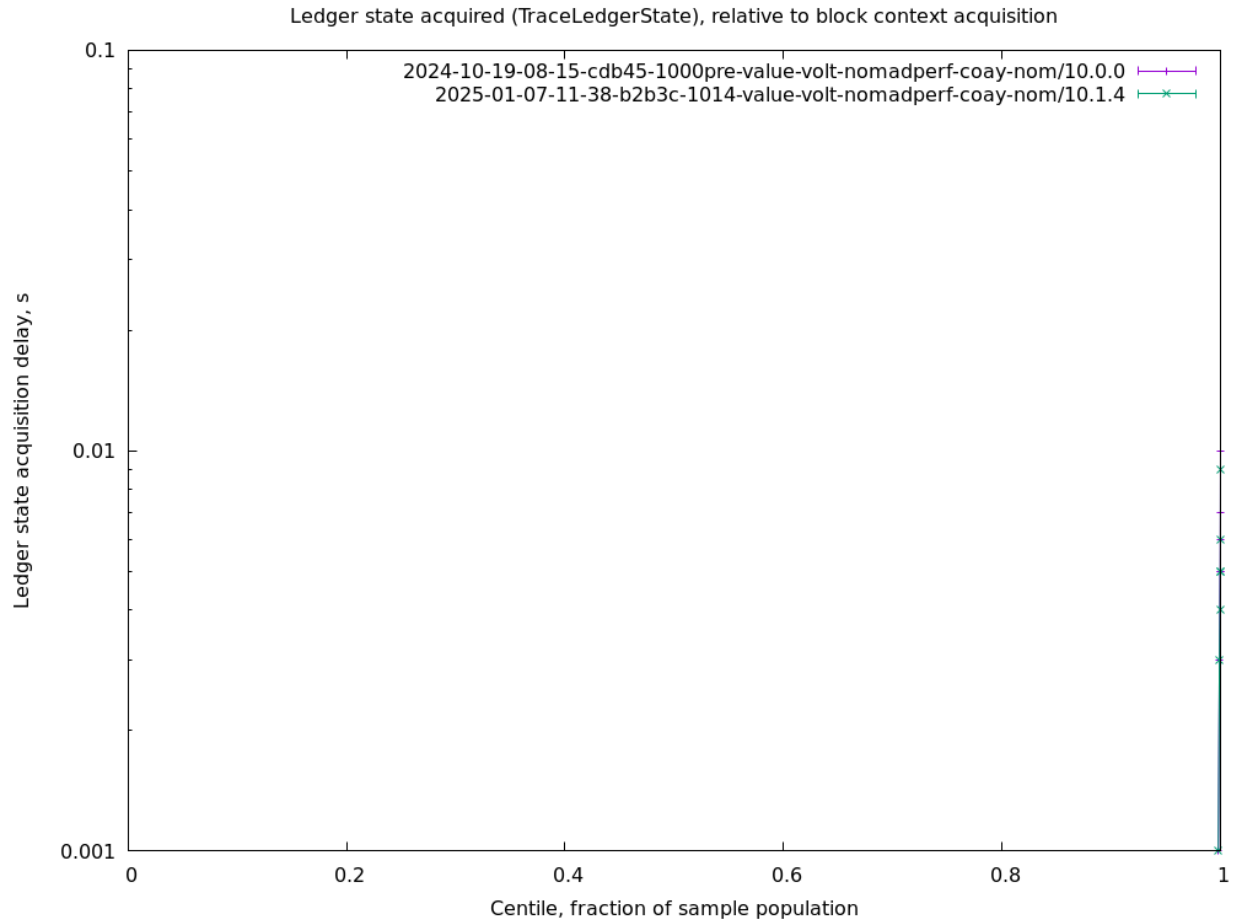

**Forge loop tardiness (cdfStarted)** Forge loop iteration start delay (TraceStartLeadershipCheck), relative to slot start



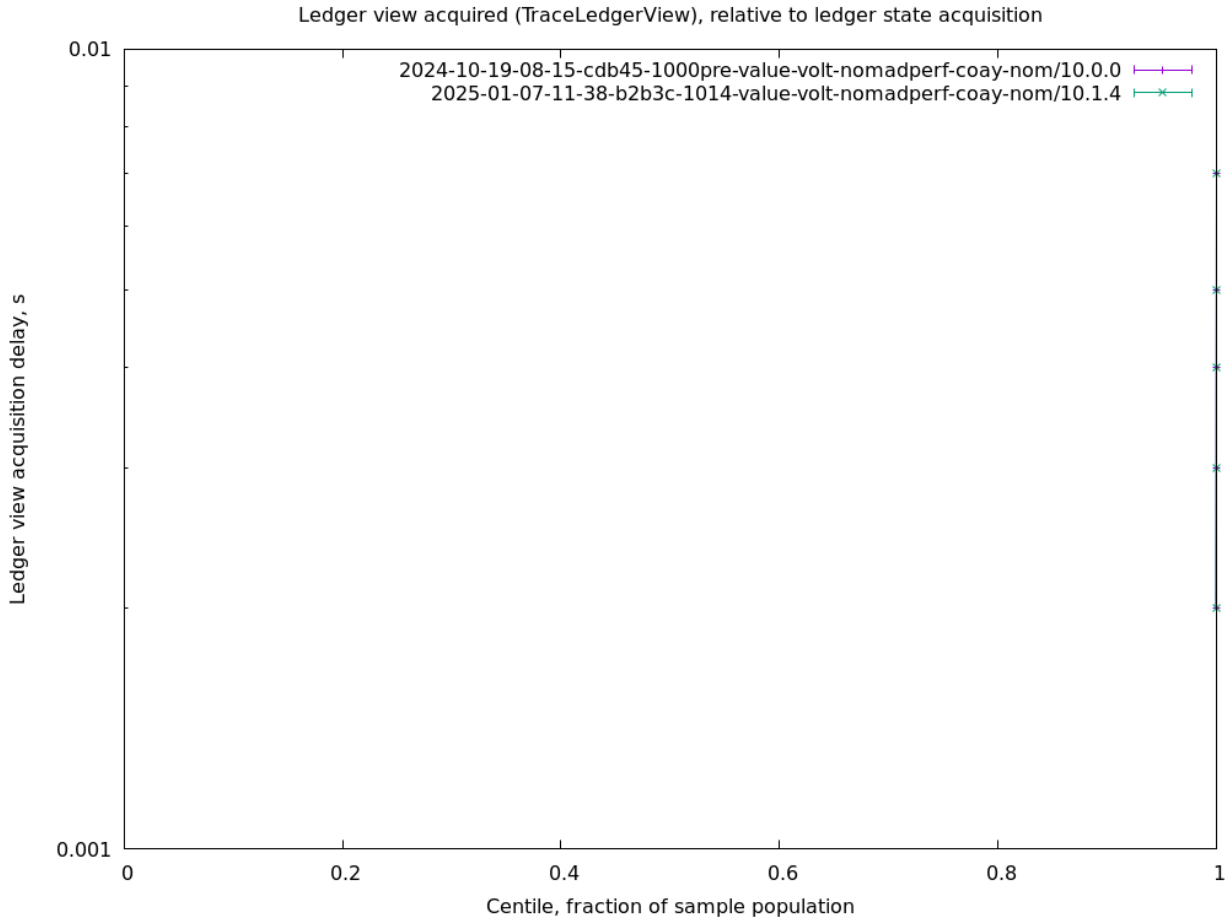

**Block context acquisition delay (cdfBlkCtx)** Block context acquired (TraceBlockContext), relative to forge loop beginning



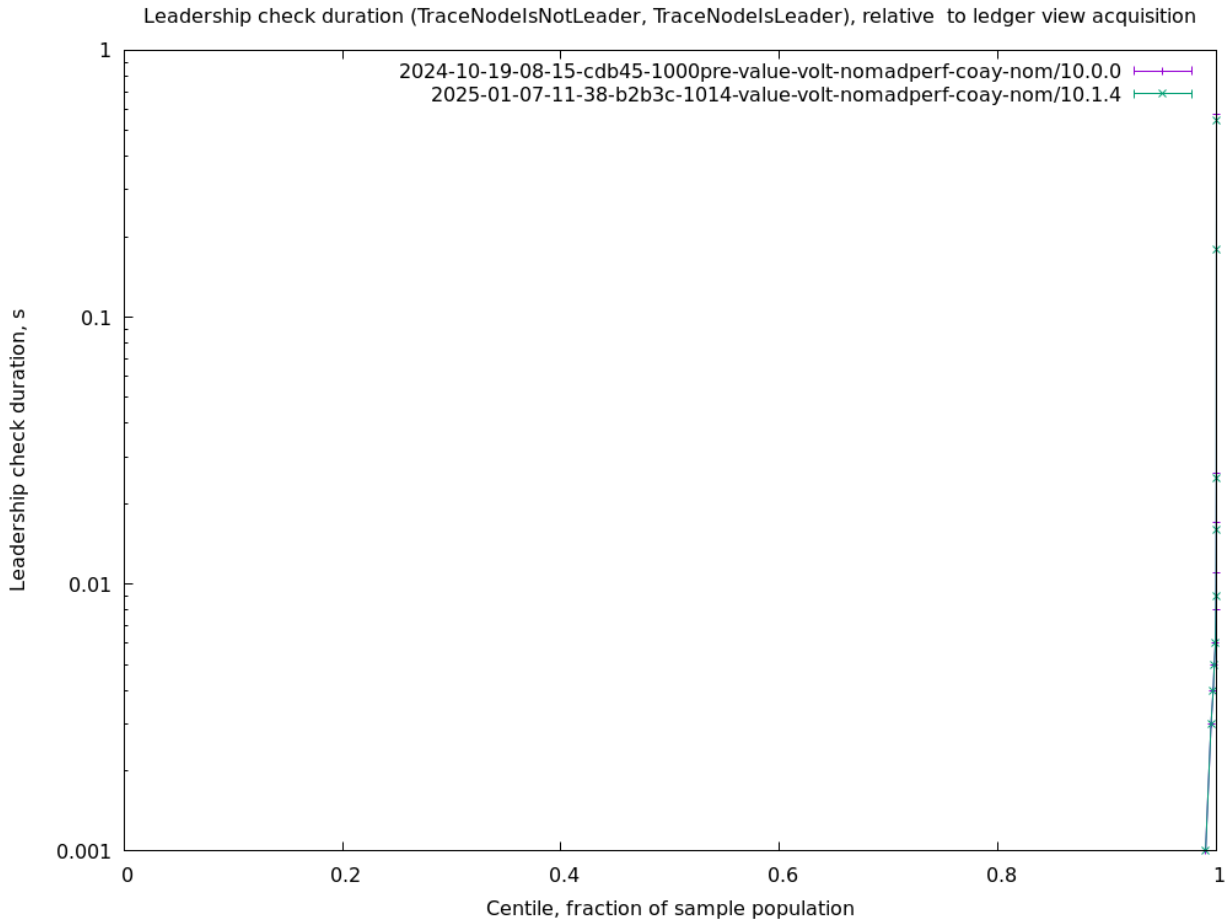
**Ledger state acquisition delay (cdfLgrState)** Ledger state acquired (TraceLedgerState), relative to block context acquisition



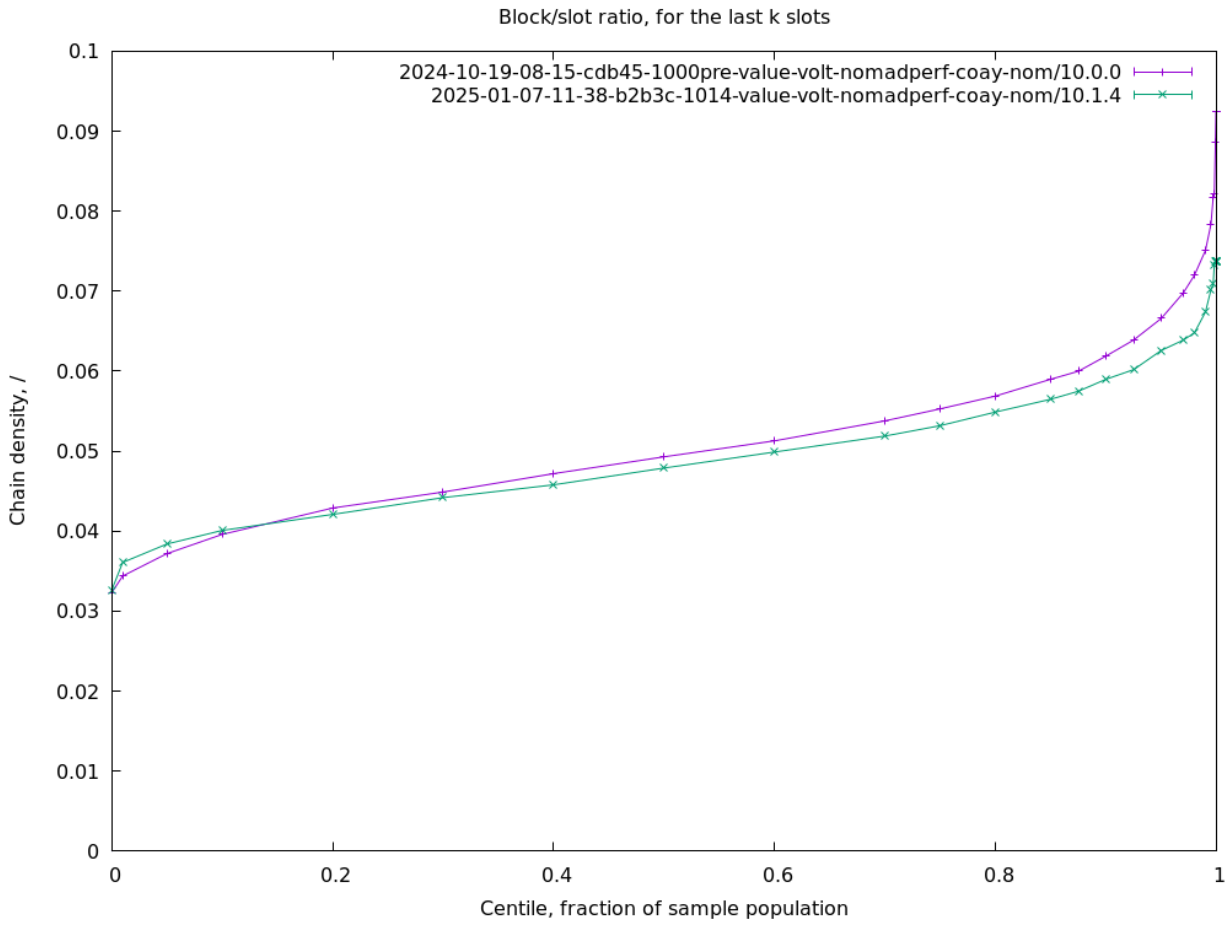

**Ledger view acquisition delay (cdfLgrView)** Ledger view acquired (TraceLedgerView), relative to ledger state acquisition



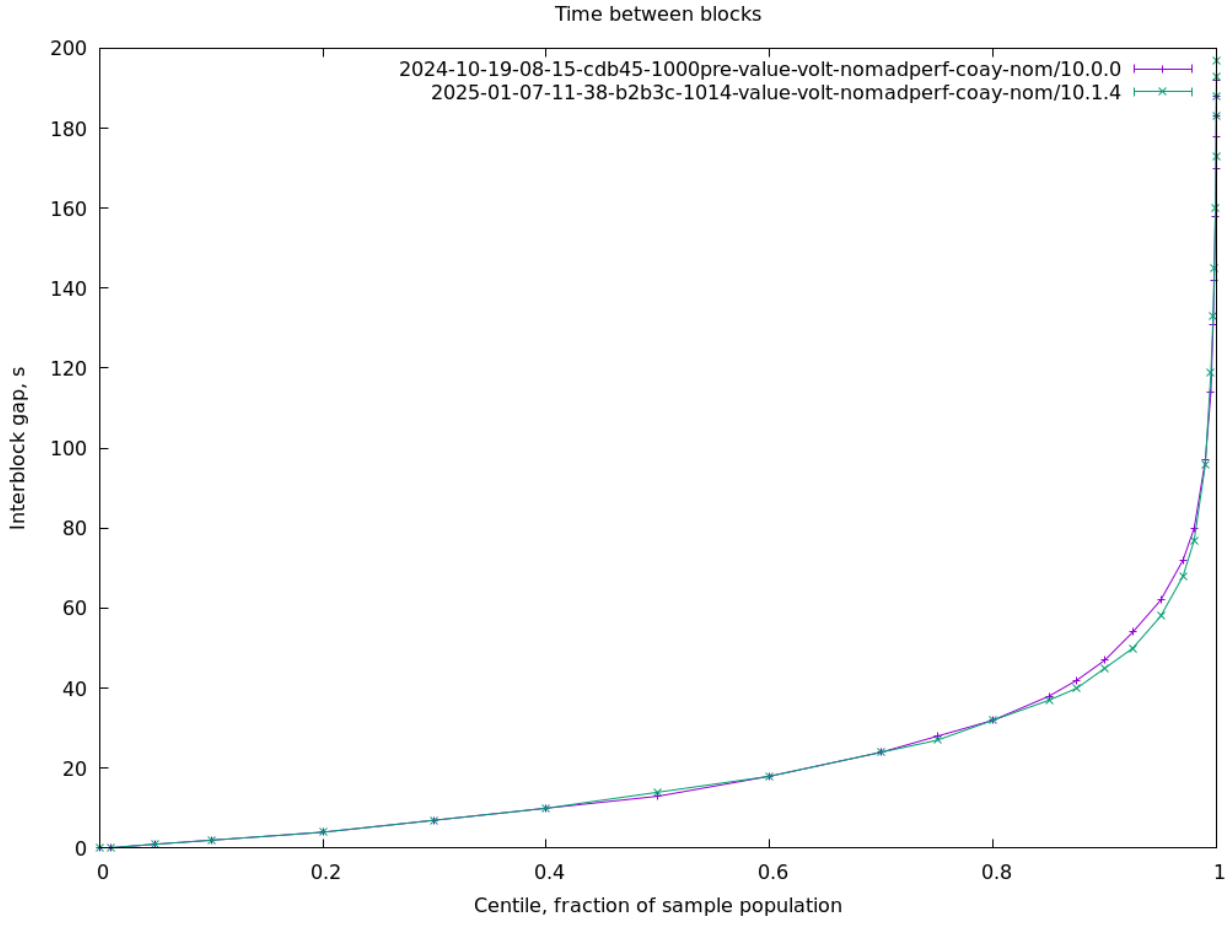

**Leadership check duration (cdfLeading)** Leadership check duration (TraceNodeIsNotLeader, TraceNodeIsLeader), relative to ledger view acquisition



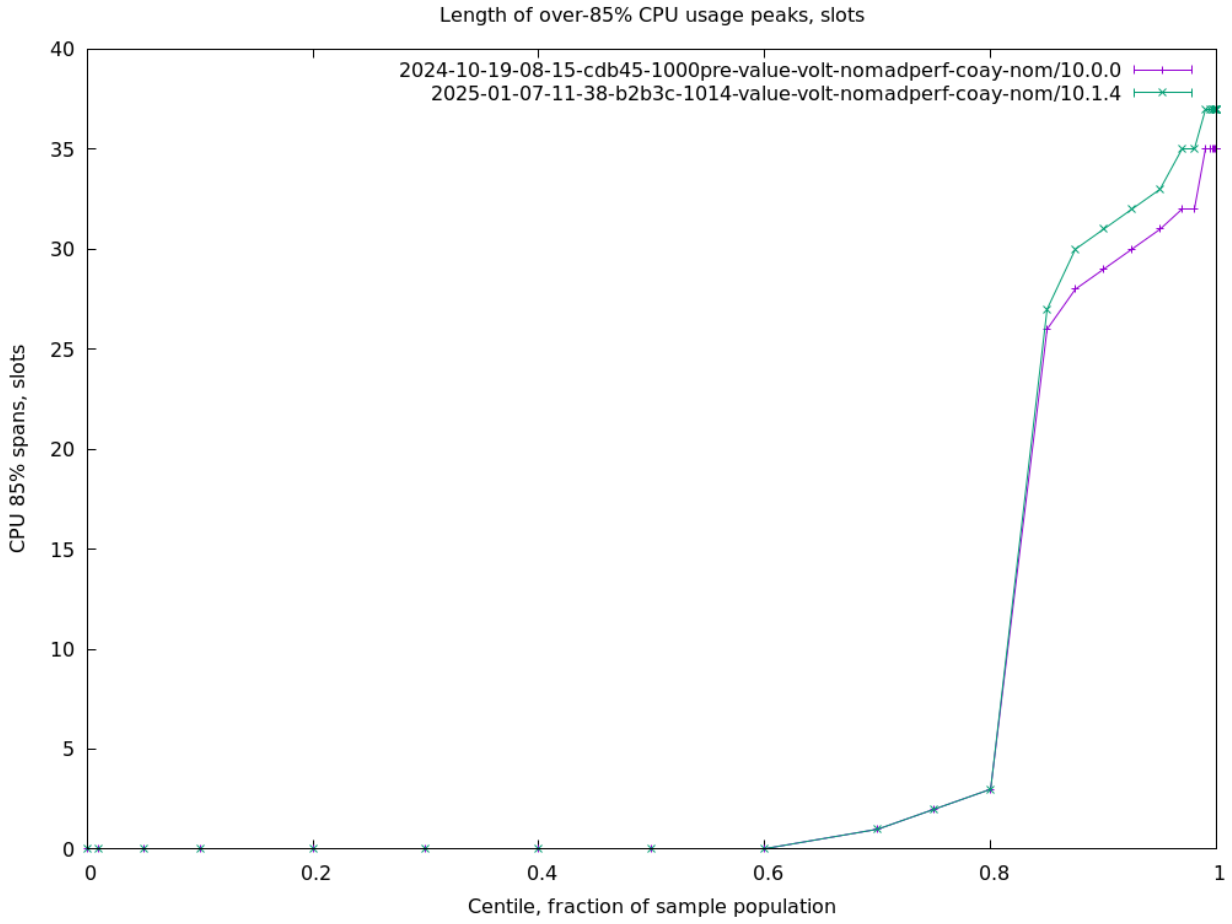

**Chain density (cdfDensity)** Block/slot ratio, for the last 'k' slots



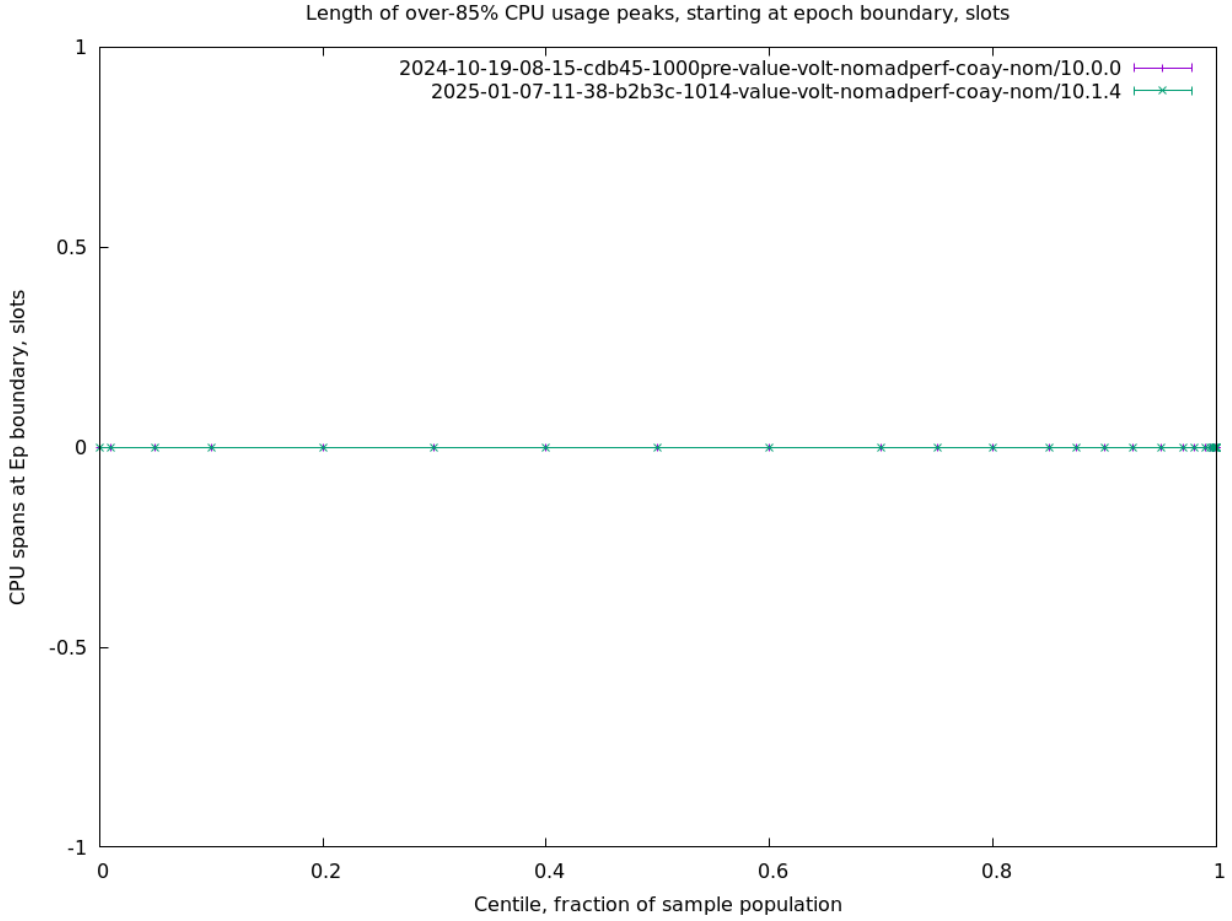

**Interblock gap (cdfBlockGap)** Time between blocks



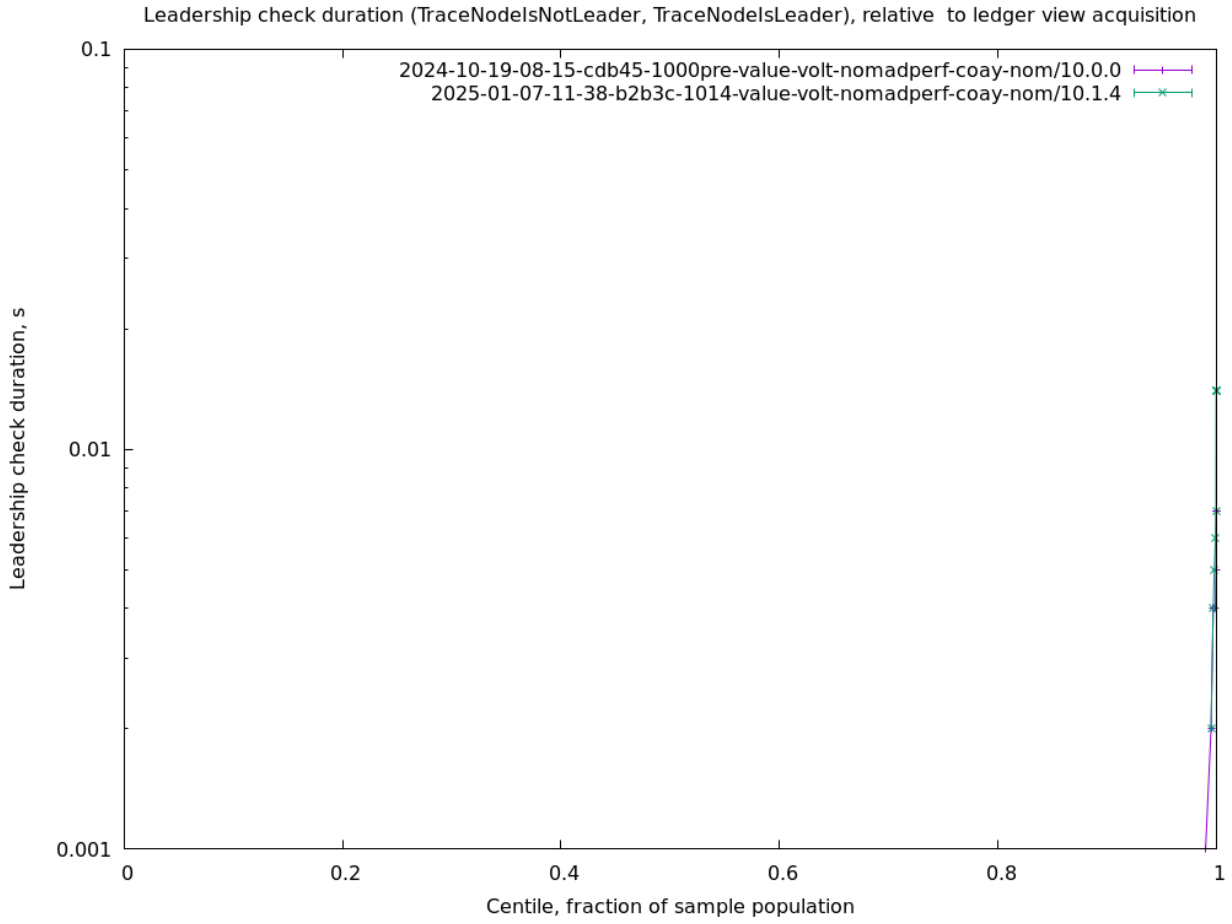

**CPU 85% spans (cdfSpanLensCpu)** Length of over-85% CPU usage peaks, slots



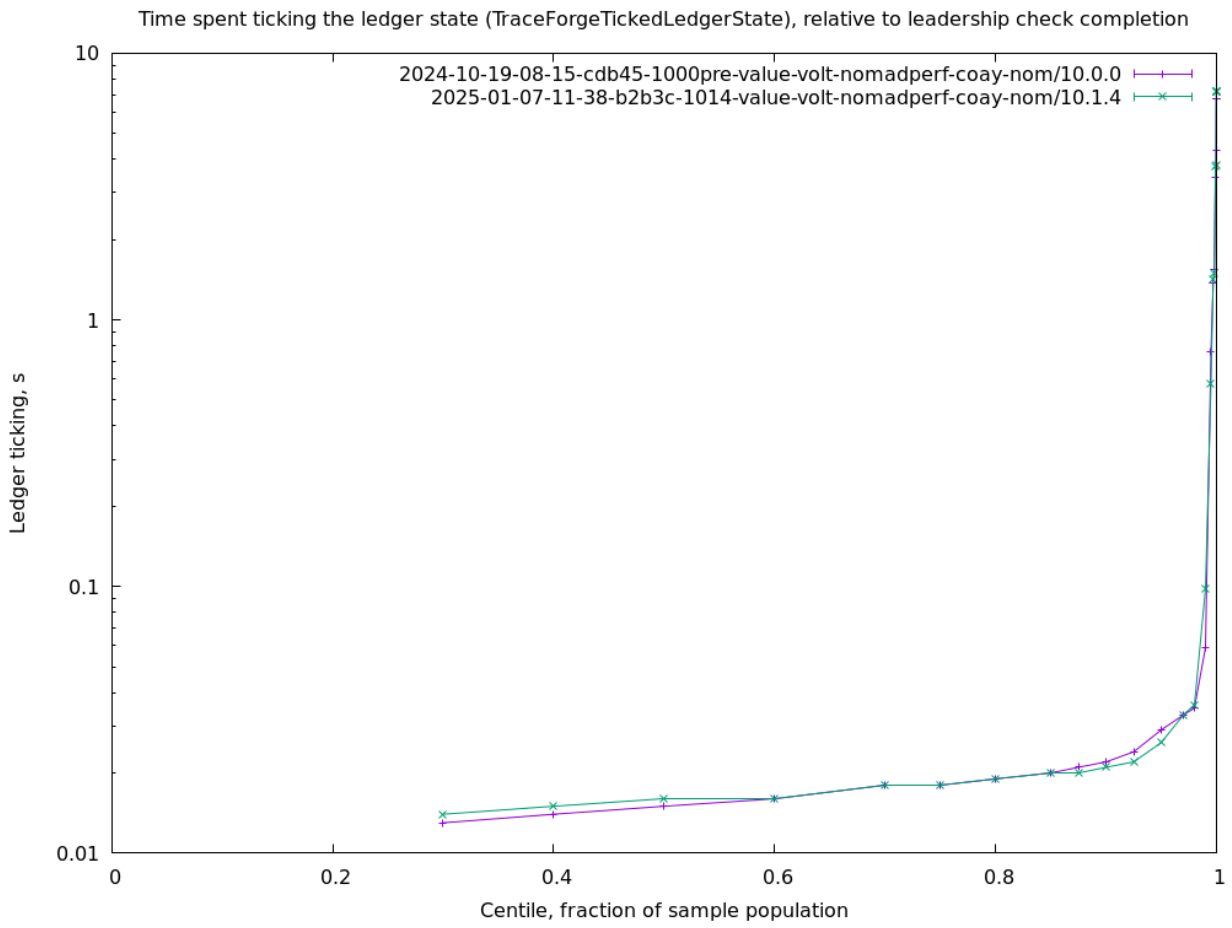

**CPU spans at Ep boundary (cdfSpanLensCpuEpoch)** Length of over-85% CPU usage peaks, starting at epoch boundary, slots



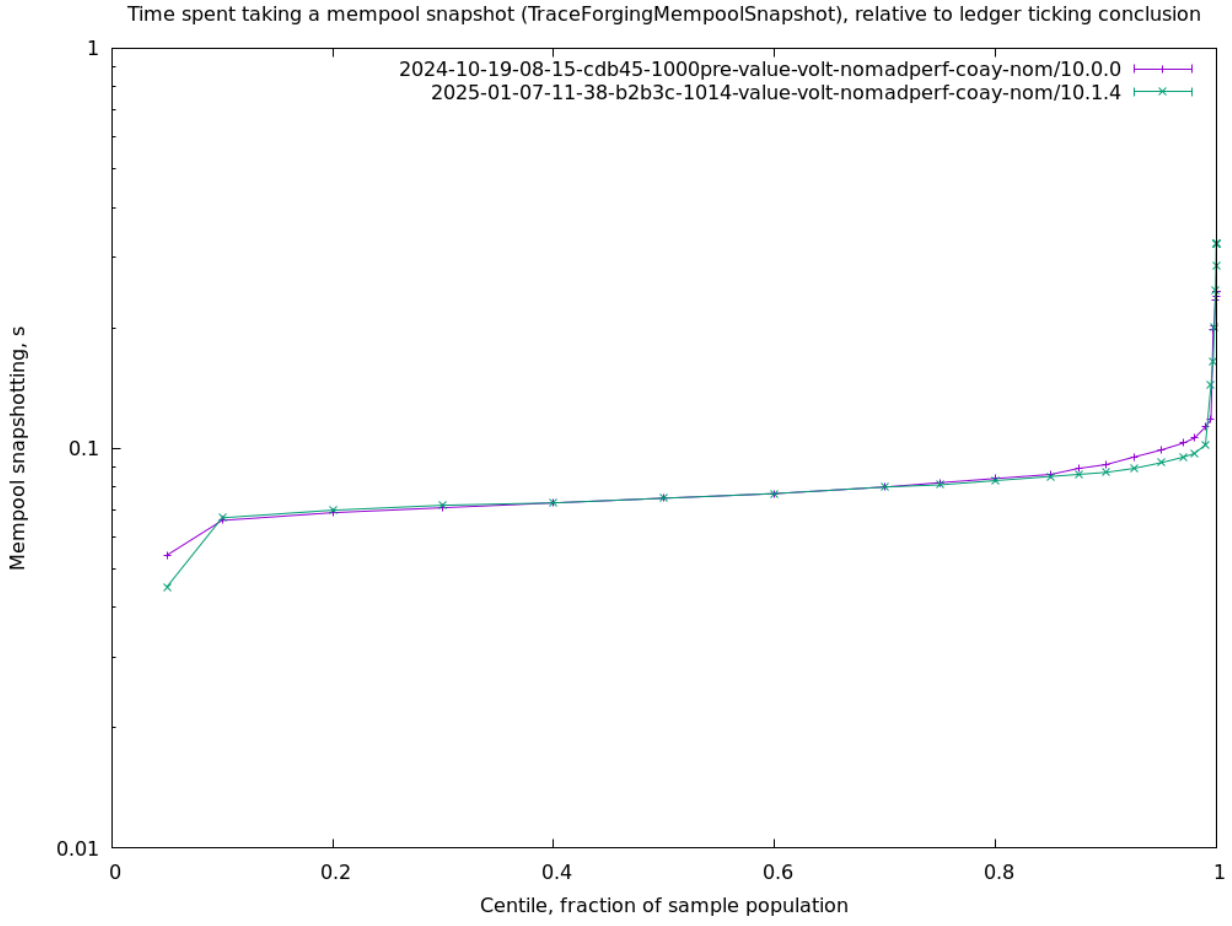
**Leadership check duration (cdfForgerLead)** Leadership check duration (TraceNodeIsNotLeader, TraceNodeIsLeader), relative to ledger view acquisition



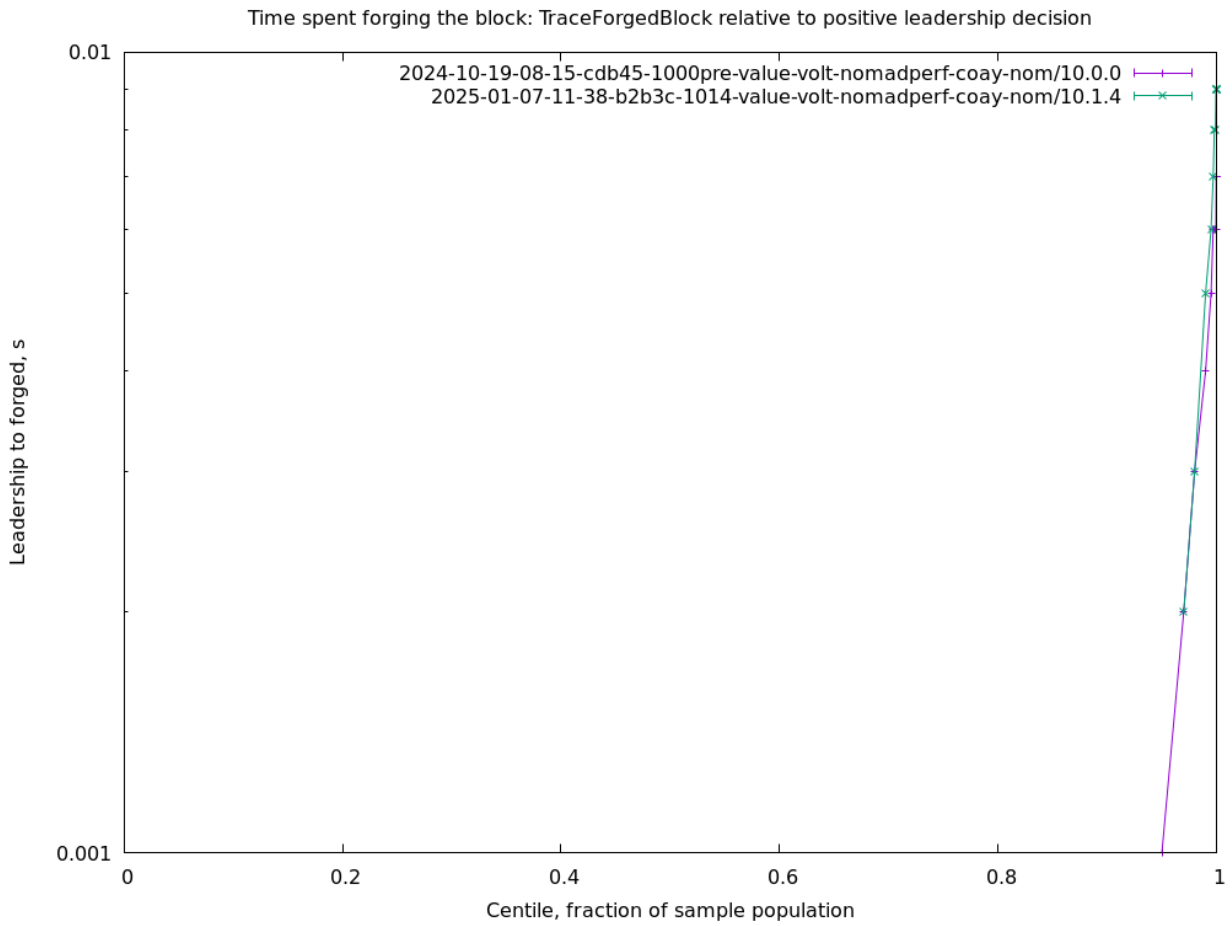

**Ledger ticking (cdfForgerTicked)** Time spent ticking the ledger state (TraceForgeTickedLedgerState), relative to leadership check completion



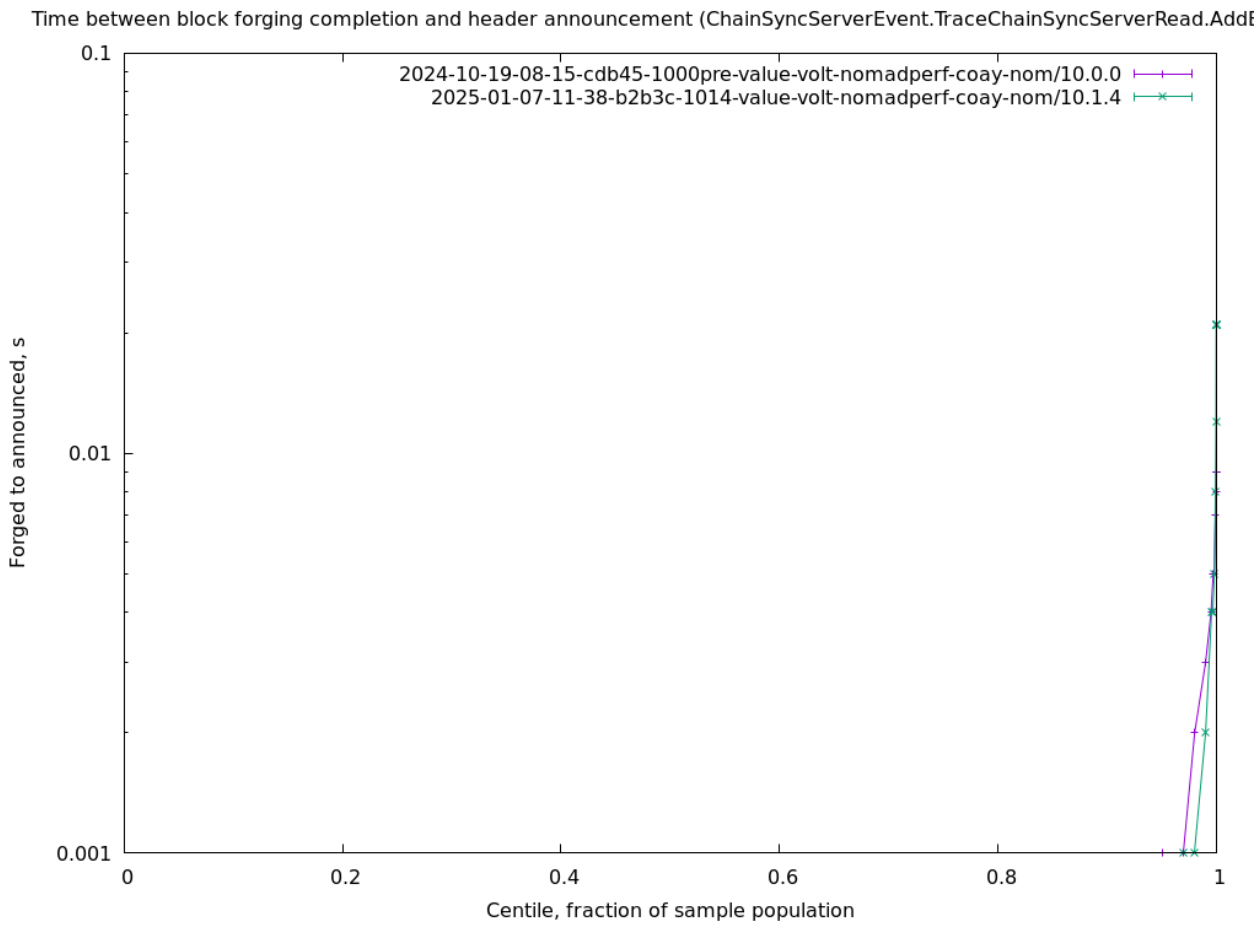

**Mempool snapshotting (cdfForgerMemSnap)** Time spent taking a mempool snapshot (TraceForgingMempool-Snapshot), relative to ledger ticking conclusion



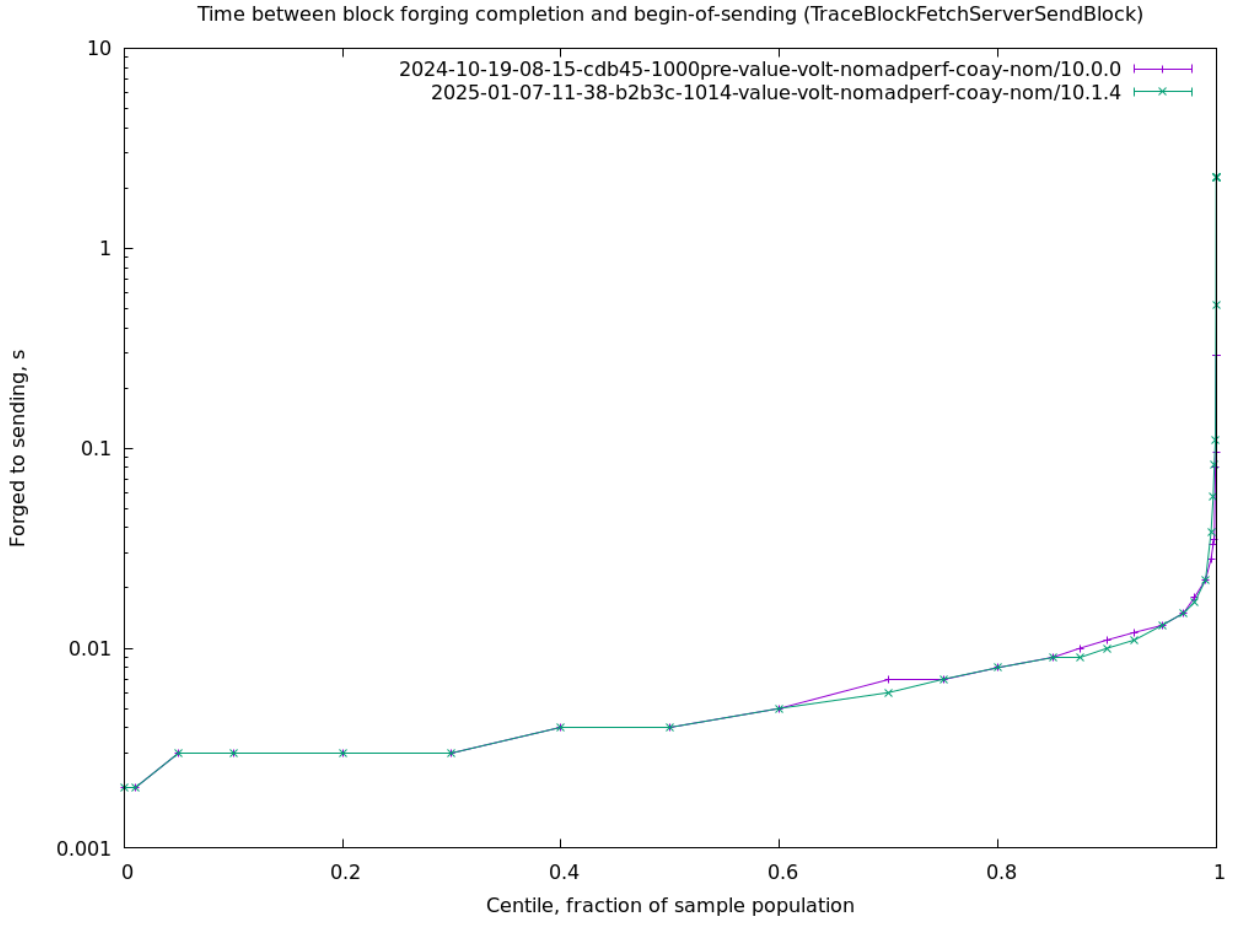
**Leadership to forged (cdfForgerForge)** Time spent forging the block: TraceForgedBlock relative to positive leadership decision



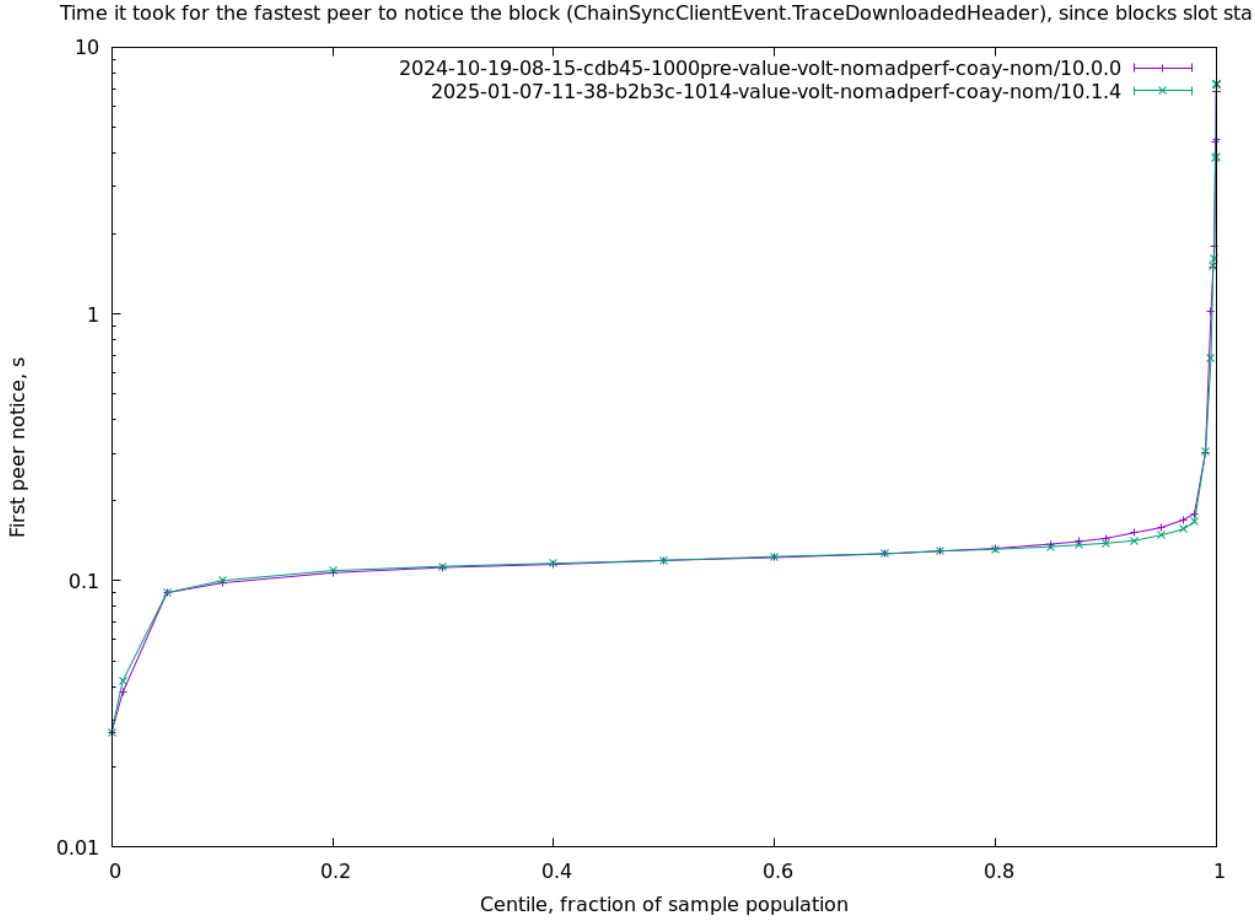

**Forged to announced (cdfForgerAnnounce)** Time between block forging completion and header announcement (ChainSyncServerEvent.TraceChainSyncServerRead.AddBlock)



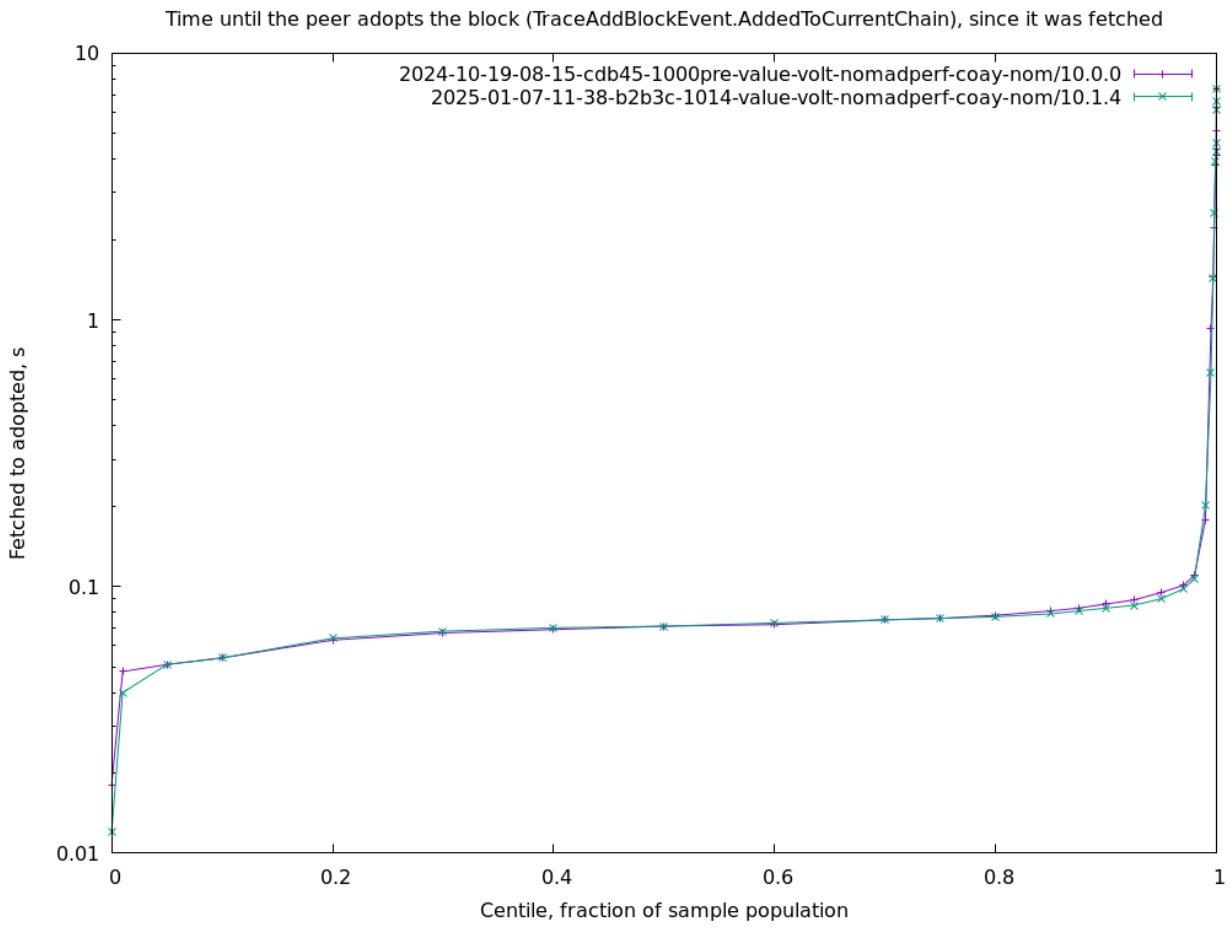

**Forged to sending (cdfForgerSend)** Time between block forging completion and begin-of-sending (TraceBlockFetchServerSendBlock)



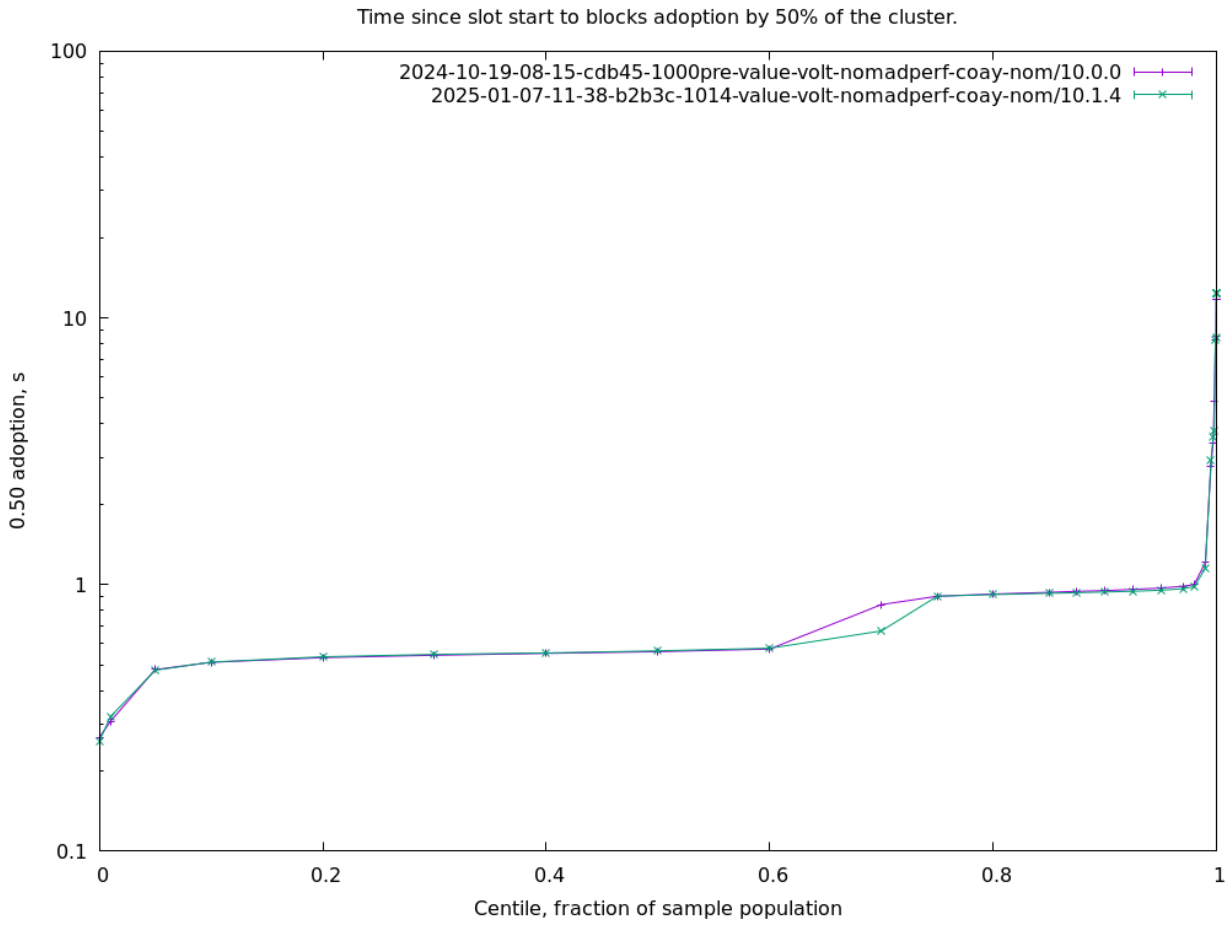
**First peer notice (cdfPeerNoticeFirst)** Time it took for the fastest peer to notice the block (ChainSyncClientEvent.TraceDownloadedHeader), since block's slot start



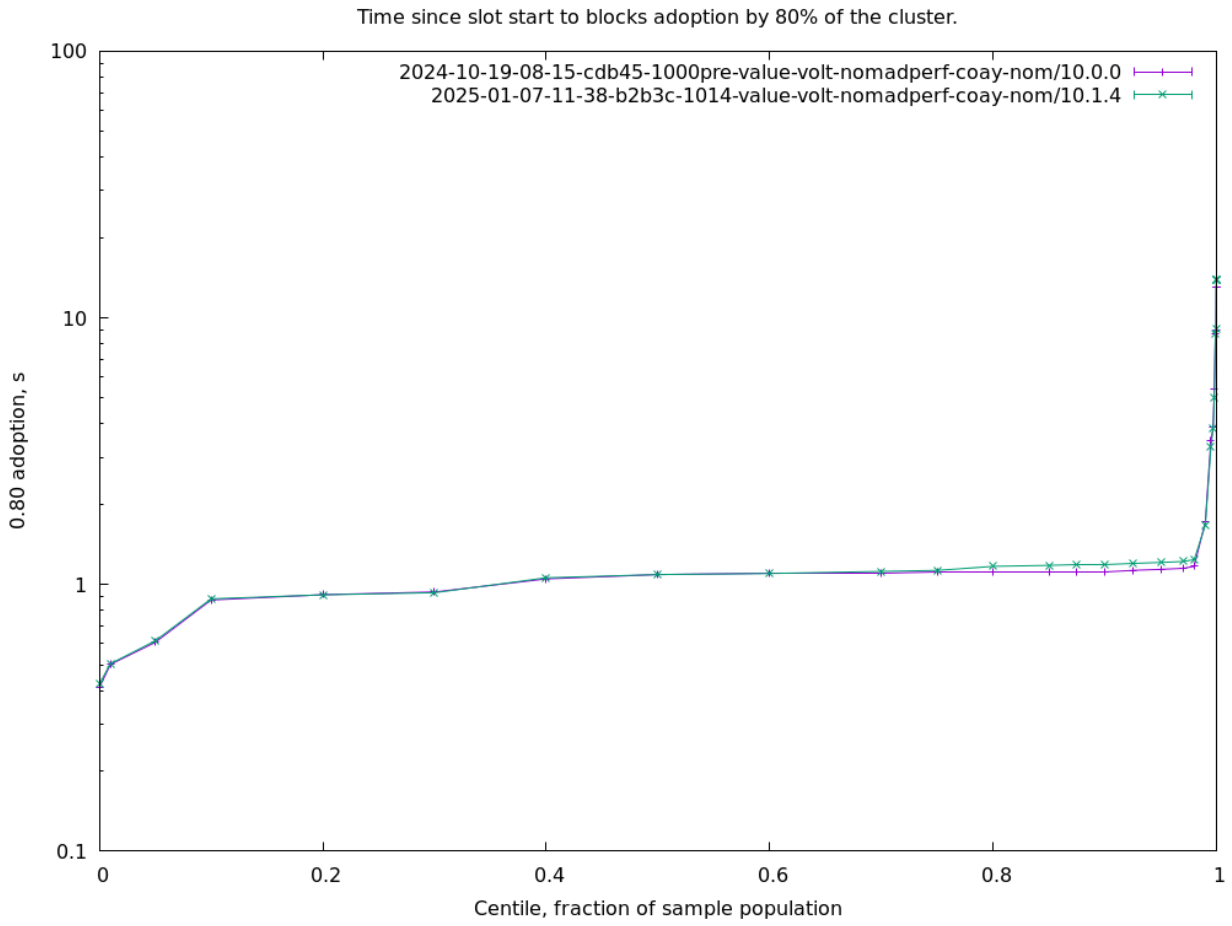

**Fetched to adopted (cdfPeerAdoption)** Time until the peer adopts the block (TraceAddBlockEvent.AddedToCurrentChain), since it was fetched



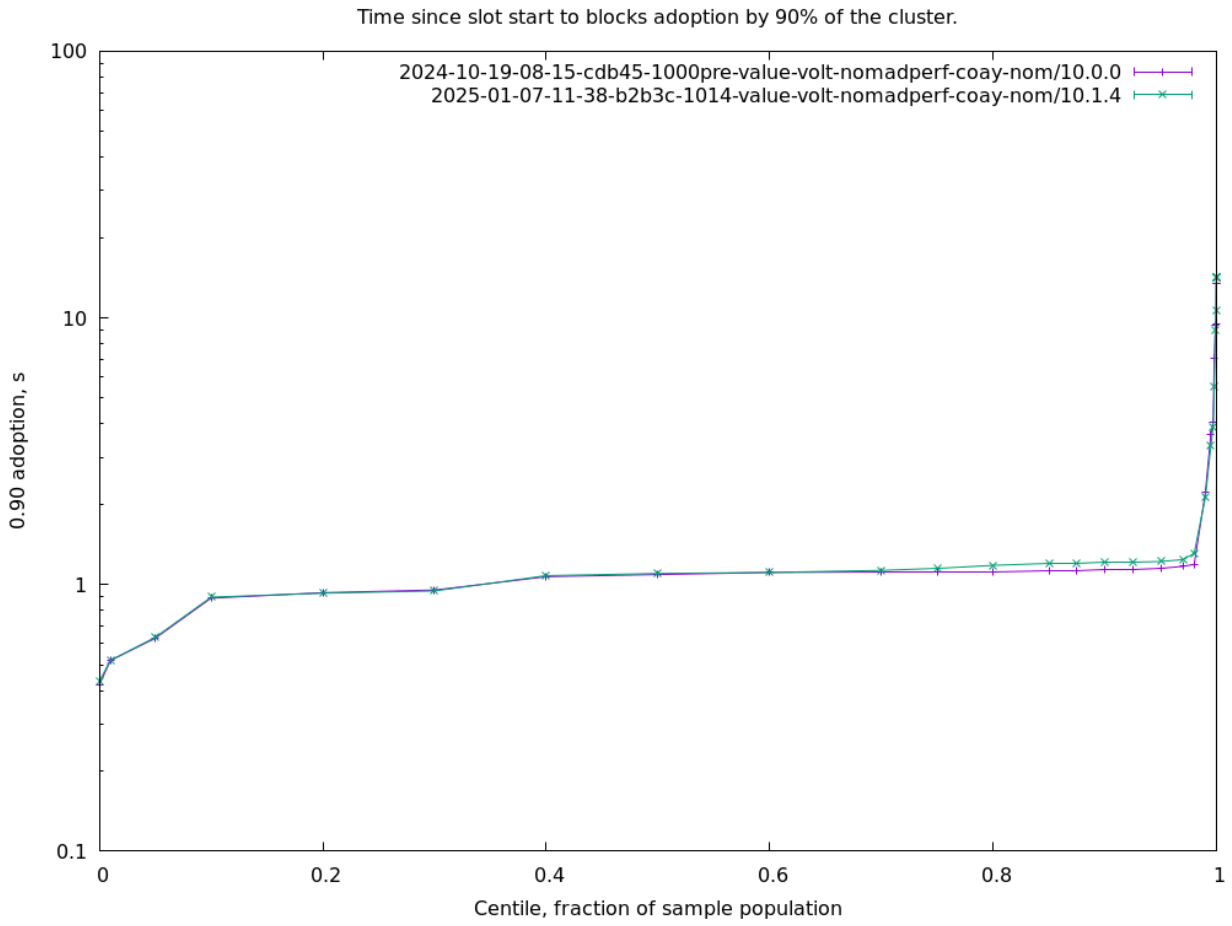

**0.50 adoption (cdf0.50)** Time since slot start to block's adoption by 50% of the cluster.



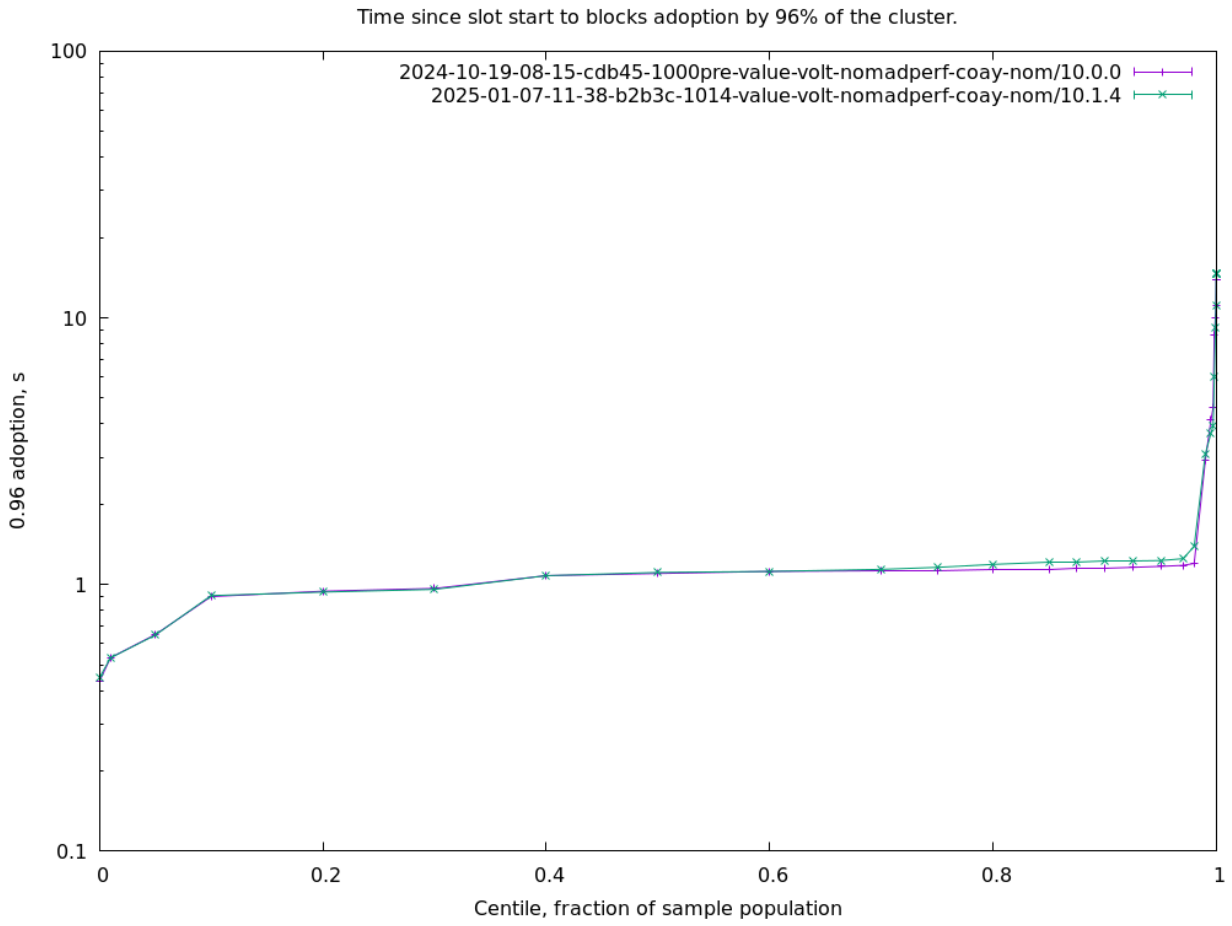

**0.80 adoption (cdf0.80)** Time since slot start to block's adoption by 80% of the cluster.




**0.90 adoption (cdf0.90)** Time since slot start to block's adoption by 90% of the cluster.




**0.96 adoption (cdf0.96)** Time since slot start to block's adoption by 96% of the cluster.


Time since slot start to blocks adoption by 96% of the cluster.
2024-10-19-08-15-cdb45-1000pre-value-volt-nomadperf-coay-nom/10.0.0
2025-01-07-11-38-b2b3c-1014-value-volt-nomadperf-coay-nom/10.1.4
0.96 adoption, s
100
10
1
0.1
0
0.2
0.4
0.6
0.8
1
Centile, fraction of sample population

# Part II

# Appendix B: data dictionary

# Chapter 4

# Block propagation metrics

**0.50 adoption (cdf0.50)** Time since slot start to block's adoption by 50% of the cluster.

**0.80 adoption (cdf0.80)** Time since slot start to block's adoption by 80% of the cluster.

**0.90 adoption (cdf0.90)** Time since slot start to block's adoption by 90% of the cluster.

**0.92 adoption (cdf0.92)** Time since slot start to block's adoption by 92% of the cluster.

**0.94 adoption (cdf0.94)** Time since slot start to block's adoption by 94% of the cluster.

**0.96 adoption (cdf0.96)** Time since slot start to block's adoption by 96% of the cluster.

**0.98 adoption (cdf0.98)** Time since slot start to block's adoption by 98% of the cluster.

**1.00 adoption (cdf1.00)** Time since slot start to block's adoption by 100% of the cluster.

**Height & slot battles (cdfBlockBattle)** For a given block, number of all abandoned blocks at its block height. Sum of height and slot battles

**Block size (cdfBlockSize)** Block size, in bytes

**Chained to forged block ratio (cdfBlocksChainedRatio)** For each host, ratio of blocks that made into chain / all forged

**Filtered to chained block ratio (cdfBlocksFilteredRatio)** For each host, ratio of blocks that passed filtering / all on chain

**Blocks per host (cdfBlocksPerHost)** For each host, number of blocks made during the entire observation period

**Forged to self-adopted (cdfForgerAdoption)** Time between block forging completion and adoption (TraceAdoptedBlock)

**Forged to announced (cdfForgerAnnounce)** Time between block forging completion and header announcement (ChainSyncServerEvent.TraceChainSyncServerRead.AddBlock)

**Slot start to announced (cdfForgerAnnounceCum)** Time since slot start until header announcement (ChainSyncServerEvent.TraceChainSyncServerRead.AddBlock)

**Acquired block context (cdfForgerBlkCtx)** Block context acquired (TraceBlockContext), relative to forge loop beginning

**Leadership to forged (cdfForgerForge)** Time spent forging the block: TraceForgedBlock relative to positive leadership decision

**Leadership check duration (cdfForgerLead)** Leadership check duration (TraceNodeIsNotLeader, TraceNodeIsLeader), relative to ledger view acquisition

**Acquired ledger state (cdfForgerLgrState)** Ledger state acquired (TraceLedgerState), relative to block context acquisition

**Acquired ledger view (cdfForgerLgrView)** Ledger view acquired (TraceLedgerView), relative to ledger state acquisition

**Mempool snapshotting (cdfForgerMemSnap)** Time spent taking a mempool snapshot (TraceForgingMempoolSnapshot), relative to ledger ticking conclusion

**Forged to sending (cdfForgerSend)** Time between block forging completion and begin-of-sending (TraceBlockFetchServerSendBlock)

**Started forge loop iteration (cdfForgerStart)** Forge loop iteration delay (TraceStartLeadershipCheck), relative to slot start

**Ledger ticking (cdfForgerTicked)** Time spent ticking the ledger state (TraceForgeTickedLedgerState), relative to leadership check completion

**Fetched to adopted (cdfPeerAdoption)** Time until the peer adopts the block (TraceAddBlockEvent.AddedToCurrentChain), since it was fetched

**Fetched to announced (cdfPeerAnnounce)** Time it took a peer to announce the block (ChainSyncServerEvent.TraceChainSy since it was fetched

**Fetch duration (cdfPeerFetch)** Time it took the peer to complete fetching the block (BlockFetchClient.CompletedBlockFetch), after having requested it

**First peer fetch (cdfPeerFetchFirst)** Time it took for the fastest peer to fetch the block (BlockFetchClient.CompletedBlockFe since block's slot start

**First peer notice (cdfPeerNoticeFirst)** Time it took for the fastest peer to notice the block (ChainSyncClientEvent.TraceDownloadedHeader), since block's slot start

**Notice to fetch request (cdfPeerRequest)** Time it took the peer to request the block body (BlockFetchClient.SendFetchRequ after it have seen its header

**Fetched to sending (cdfPeerSend)** Time until the peer started sending the block (BlockFetchServer.SendBlock), since it was fetched

# Chapter 5

# Cluster performance metrics

**RTS alloc rate (Alloc)** RTS-reported allocation rate, MB/sec

**Process CPU usage (CentiCpu)** Kernel-reported CPU process usage, % of a single core

**RTS GC CPU usage (CentiGC)** RTS-reported GC CPU usage, % of a single core

**RTS Mutator CPU usage (CentiMut)** RTS-reported mutator CPU usage, % of a single core

**Filesystem reads (FsRd)** Number of bytes which this process really did cause to be fetched from the storage layer, per second

**Filesystem writes (FsWr)** Number of bytes which this process caused to be sent to the storage layer, modulo truncate(), per second

**Major GCs (GcsMajor)** Major garbage collection RTS events

**Minor GCs (GcsMinor)** Minor garbage collection RTS events

**RTS heap size (Heap)** RTS-reported heap size, MB

**RTS live GC dateset (Live)** RTS-reported GC live data size, MB

**Network reads (NetRd)** Network reads, kB/sec

**Network writes (NetWr)** Network writes, kB/sec

**Kernel RSS (RSS)** Kernel-reported RSS (Resident Set Size) of the process, MB

**Block context acquisition delay (cdfBlkCtx)** Block context acquired (TraceBlockContext), relative to forge loop beginning

**Interblock gap (cdfBlockGap)** Time between blocks

**Chain density (cdfDensity)** Block/slot ratio, for the last 'k' slots

**Leadership check duration (cdfLeading)** Leadership check duration (TraceNodeIsNotLeader, TraceNodeIsLeader), relative to ledger view acquisition

**Ledger state acquisition delay (cdfLgrState)** Ledger state acquired (TraceLedgerState), relative to block context acquisition

**Ledger view acquisition delay (cdfLgrView)** Ledger view acquired (TraceLedgerView), relative to ledger state acquisition

**CPU 85% spans (cdfSpanLensCpu)** Length of over-85% CPU usage peaks, slots

**CPU spans at Ep boundary (cdfSpanLensCpuEpoch)** Length of over-85% CPU usage peaks, starting at epoch boundary, slots

**Forge loop tardiness (cdfStarted)** Forge loop iteration start delay (TraceStartLeadershipCheck), relative to slot start

**Forge loop starts (cdfStarts)** For any given slot, how many forging loop starts were registered