Reads/0B

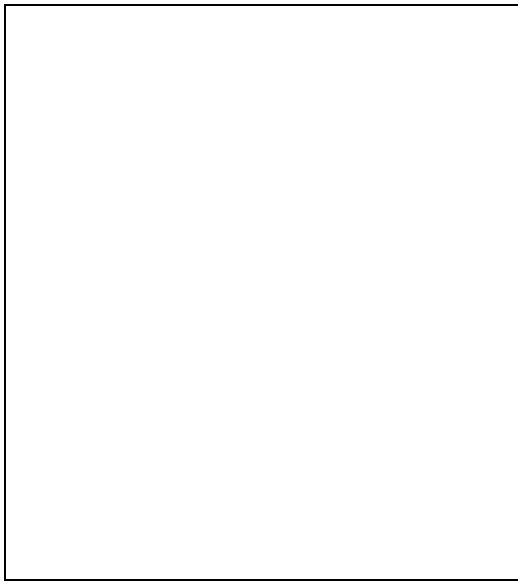
Violin Plot

This chart shows the relationship between function/parameter and iteration time. The thickness of the shaded region indicates the probability that a measurement of the given function/parameter would take a particular length of time.

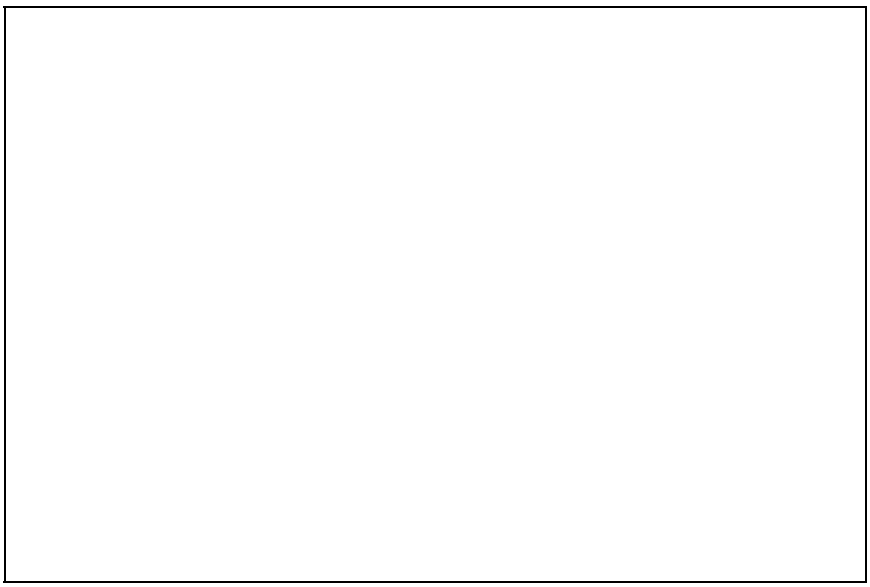
Line Chart

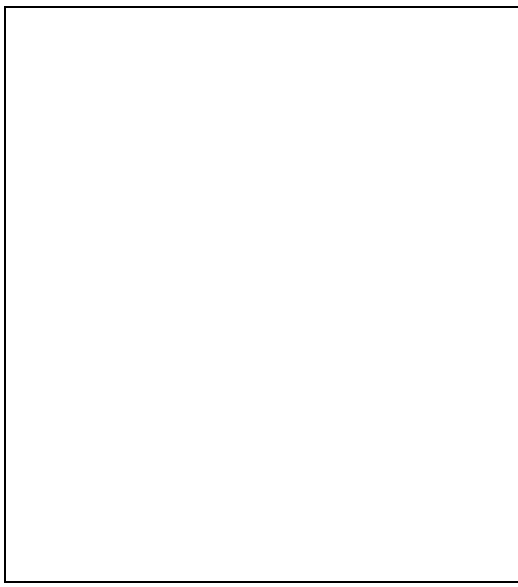
This chart shows the mean measured time for each function as the input (or the size of the input) increases.

Reads/minio: one-node/0B

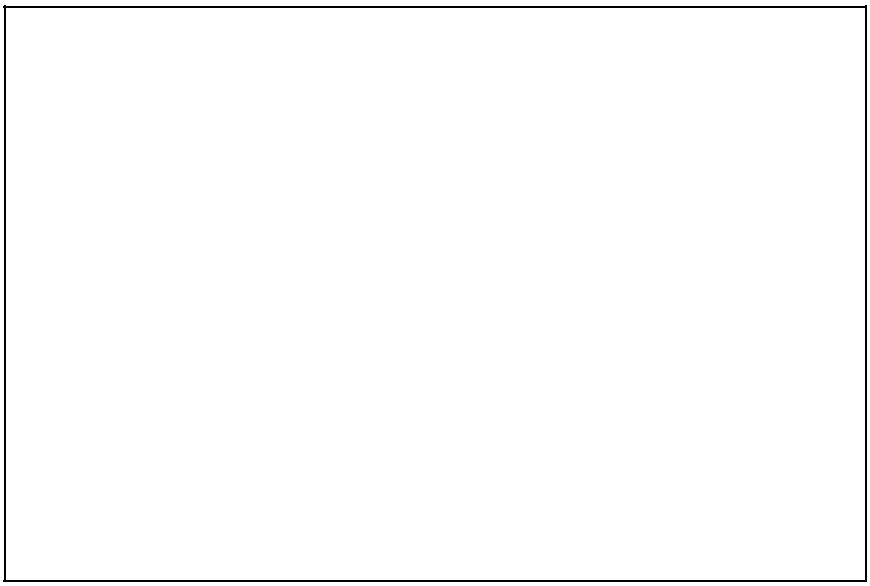


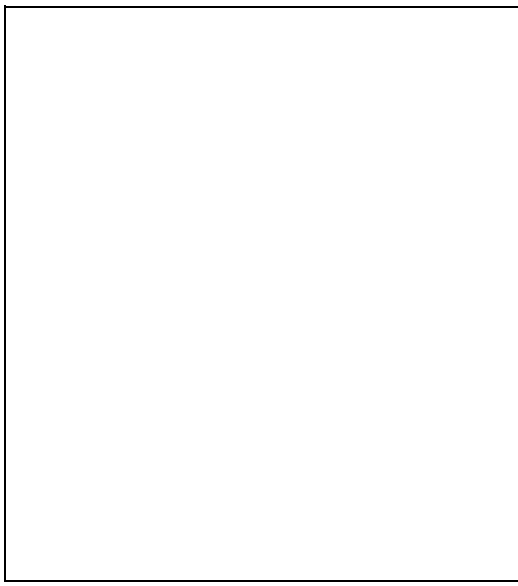
Reads/minio: two-node/0B



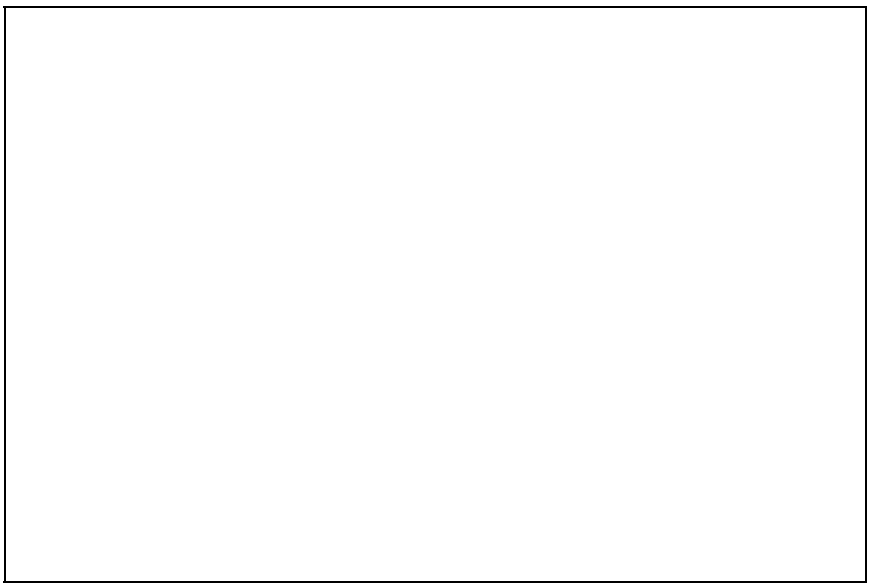


Reads/minio: one-node/0B #2

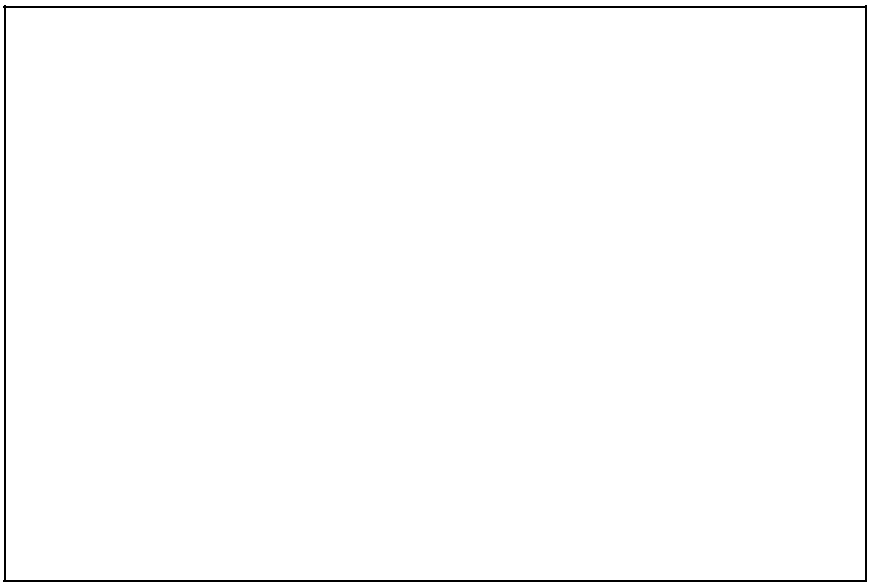




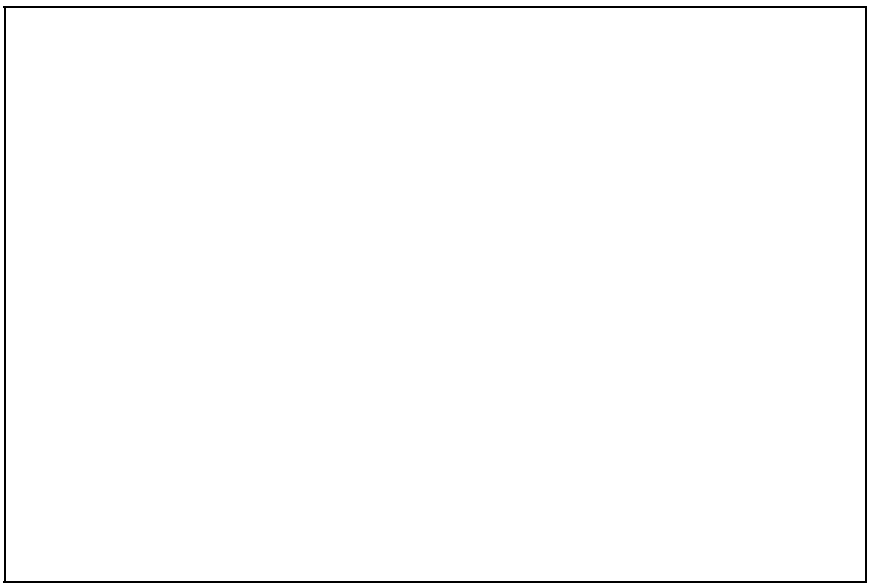
Reads/minio: two-node/0B #2



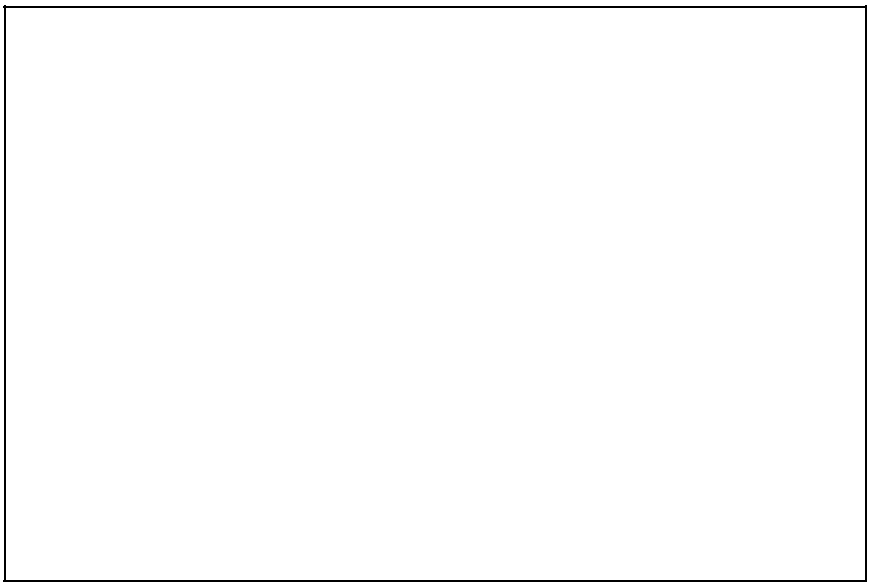
Reads/minio: one-node/0B #3



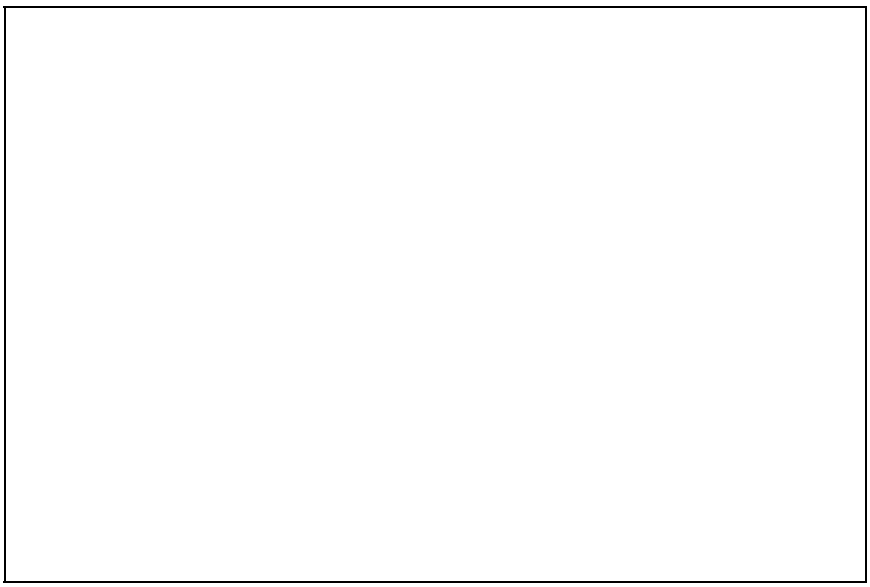
Reads/minio: two-node/0B #3

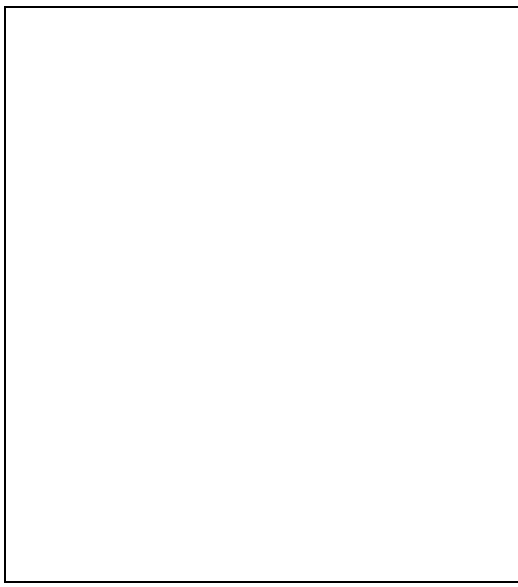


Reads/minio: one-node/0B #4



Reads/minio: two-node/0B #4





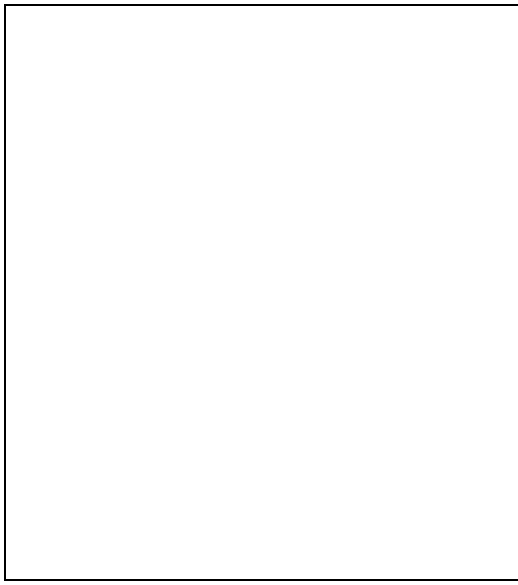
This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Reads/100MiB

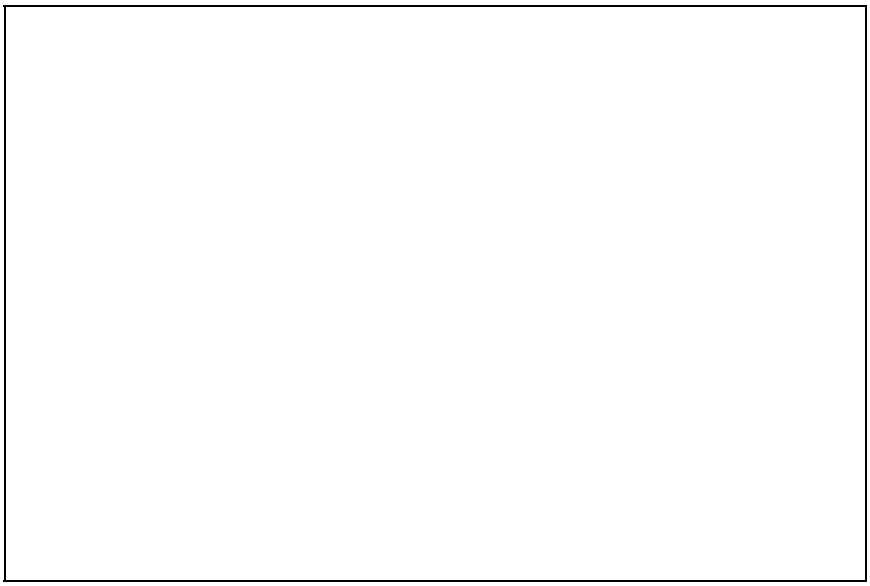
Violin Plot

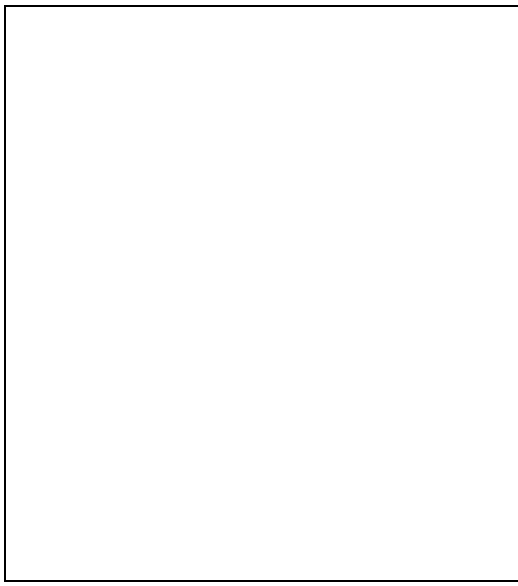
This chart shows the relationship between function/parameter and iteration time. The thickness of the shaded region indicates the probability that a measurement of the given function/parameter would take a particular length of time.

Reads/bob: one-node/100MiB

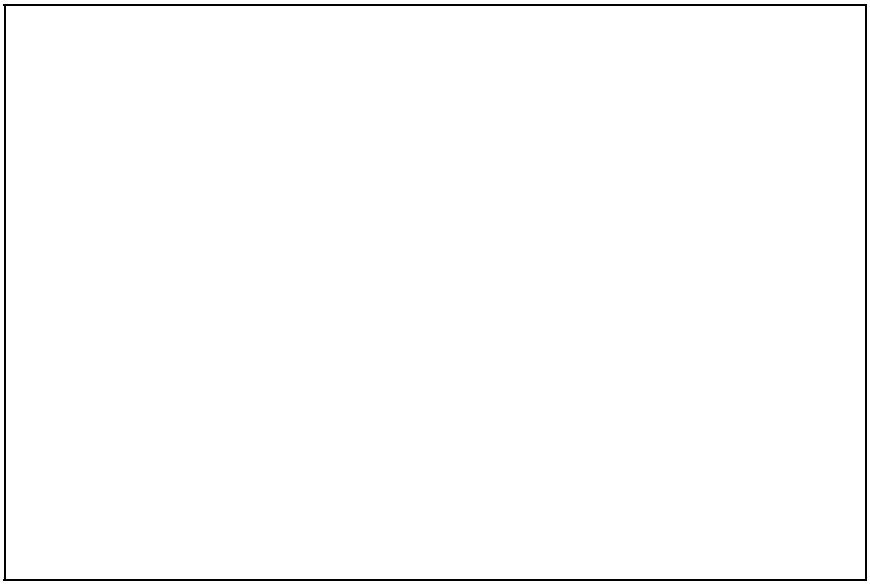


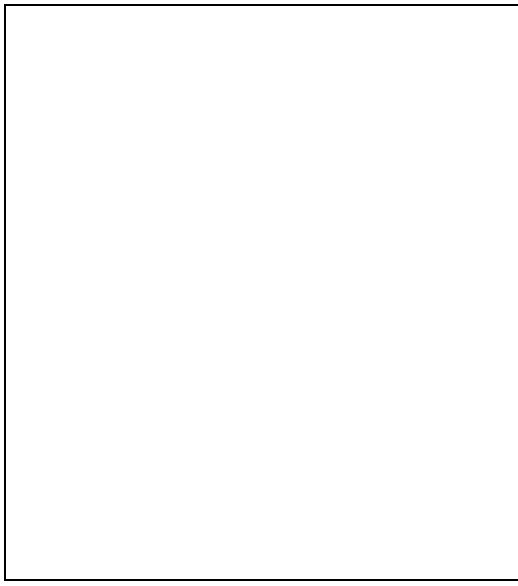
Reads/minio: one-node/100MiB



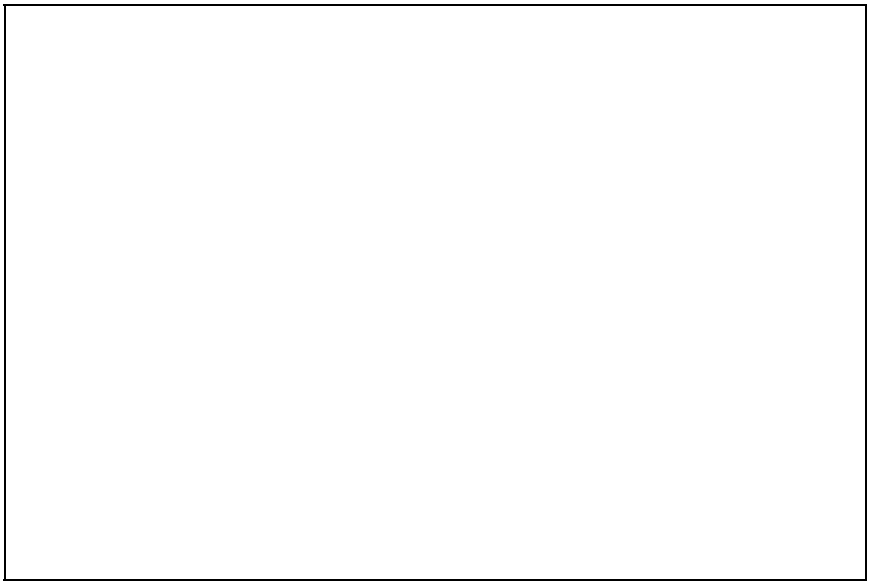


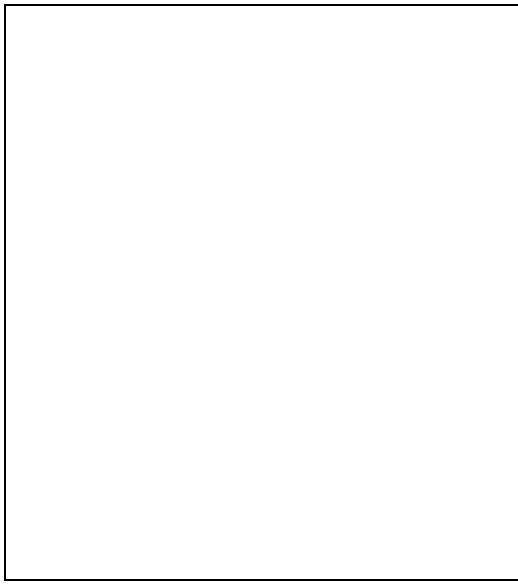
Reads/bob: two-node/100MiB





Reads/minio: two-node/100MiB





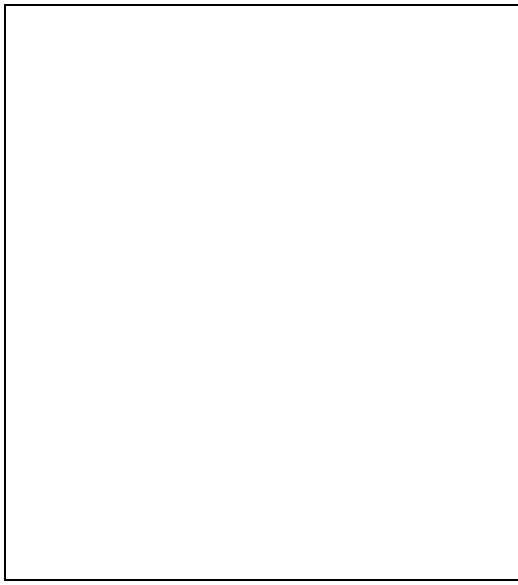
This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Reads/1KiB

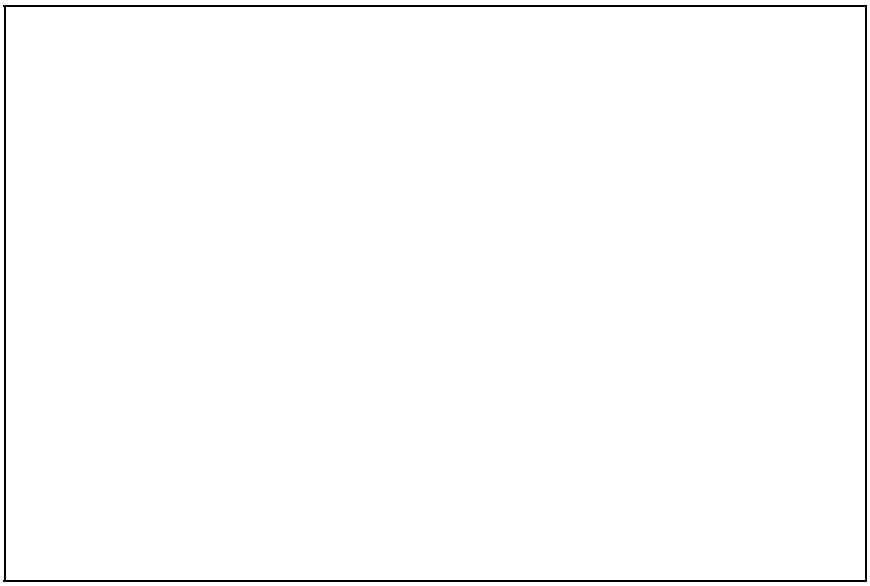
Violin Plot

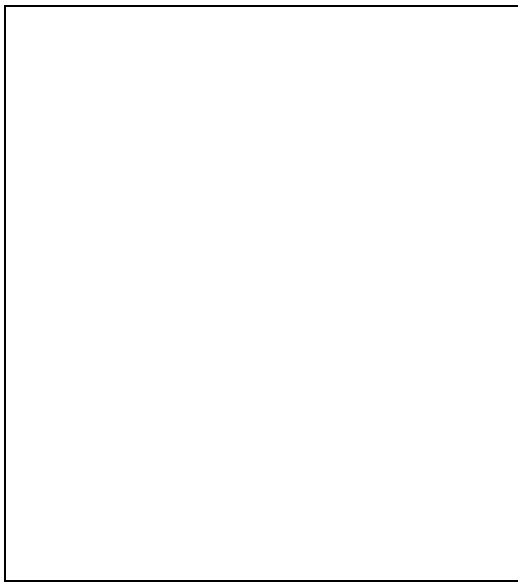
This chart shows the relationship between function/parameter and iteration time. The thickness of the shaded region indicates the probability that a measurement of the given function/parameter would take a particular length of time.

Reads/bob: one-node/1KiB

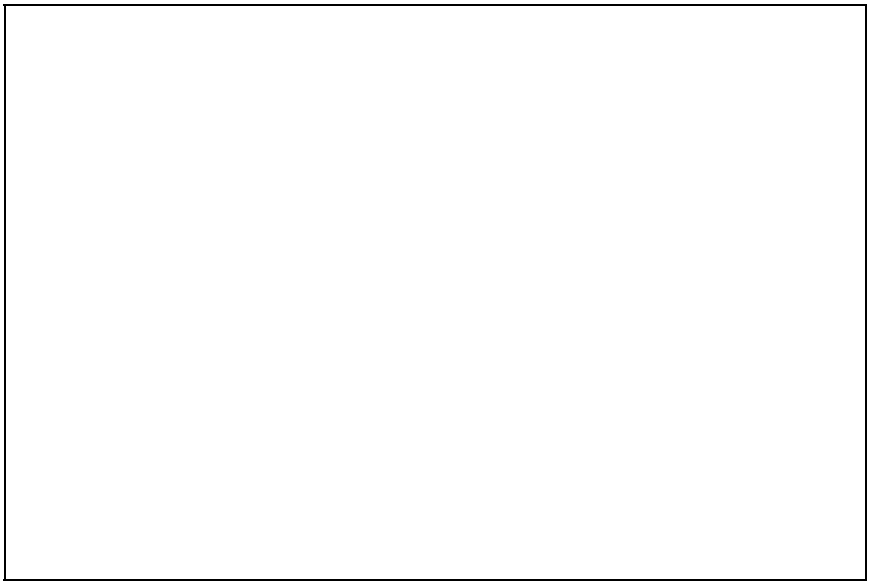


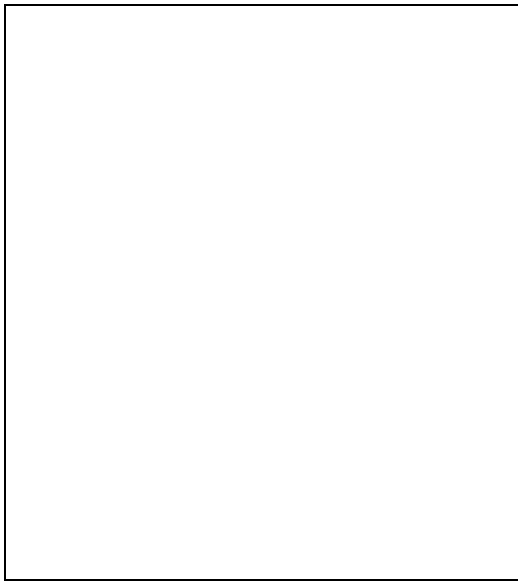
Reads/minio: one-node/1KiB



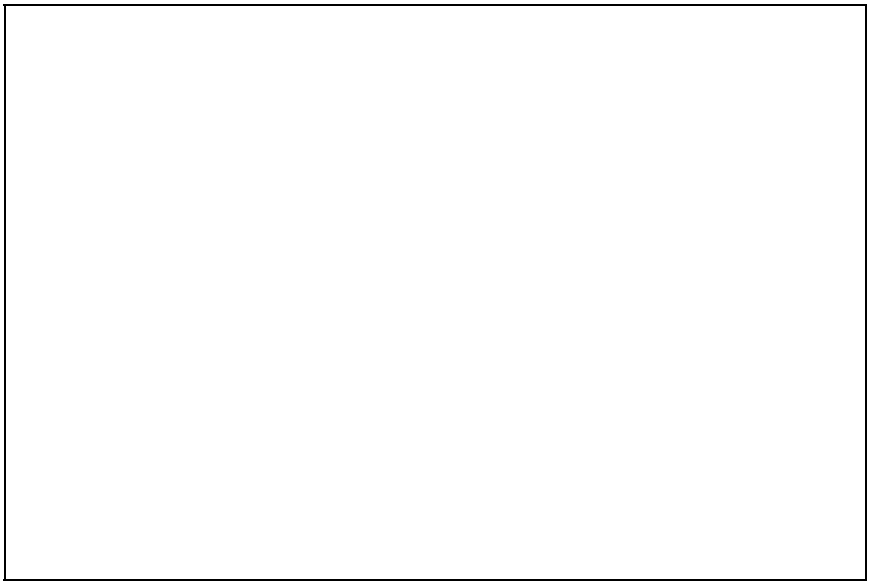


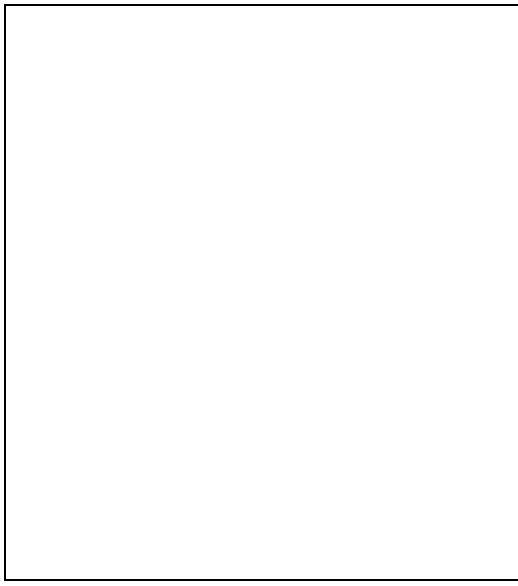
Reads/bob: two-node/1KiB





Reads/minio: two-node/1KiB





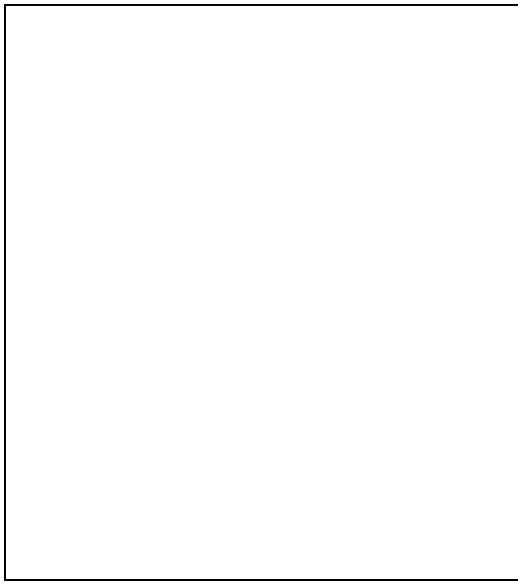
This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Reads/1MiB

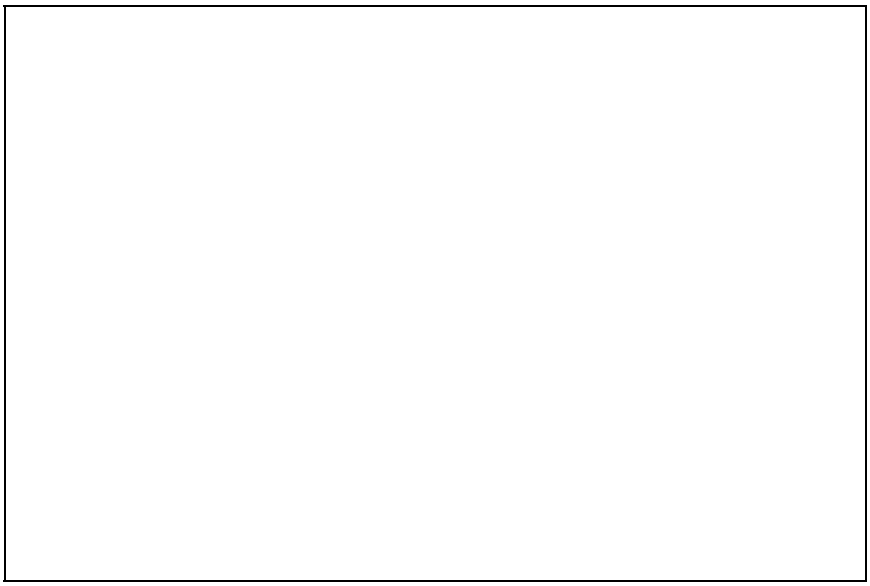
Violin Plot

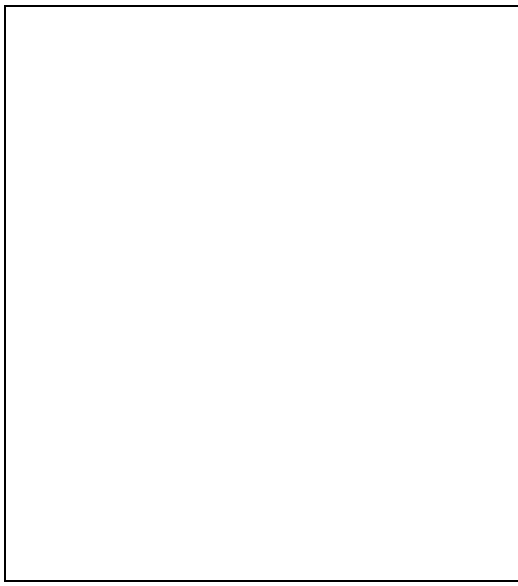
This chart shows the relationship between function/parameter and iteration time. The thickness of the shaded region indicates the probability that a measurement of the given function/parameter would take a particular length of time.

Reads/bob: one-node/1MiB

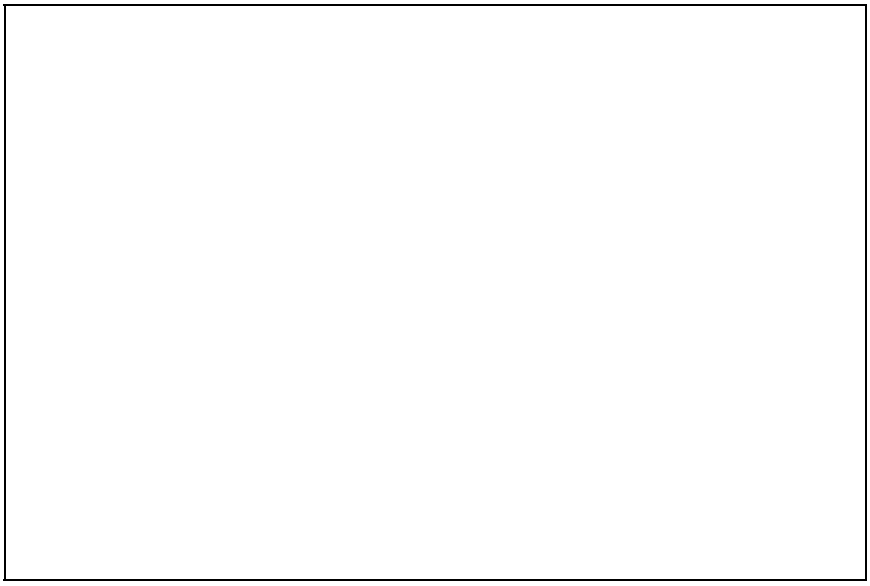


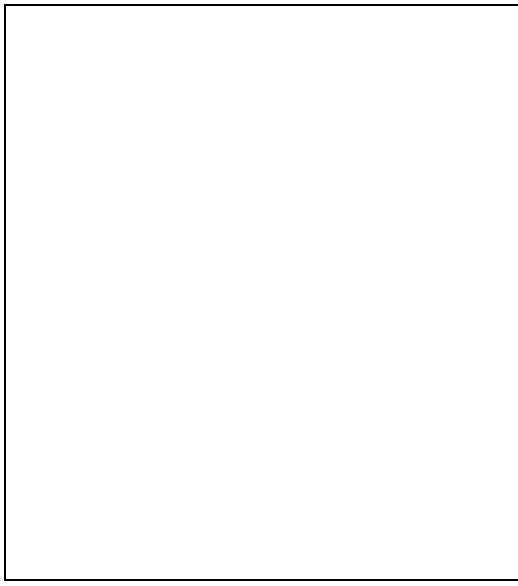
Reads/minio: one-node/1MiB



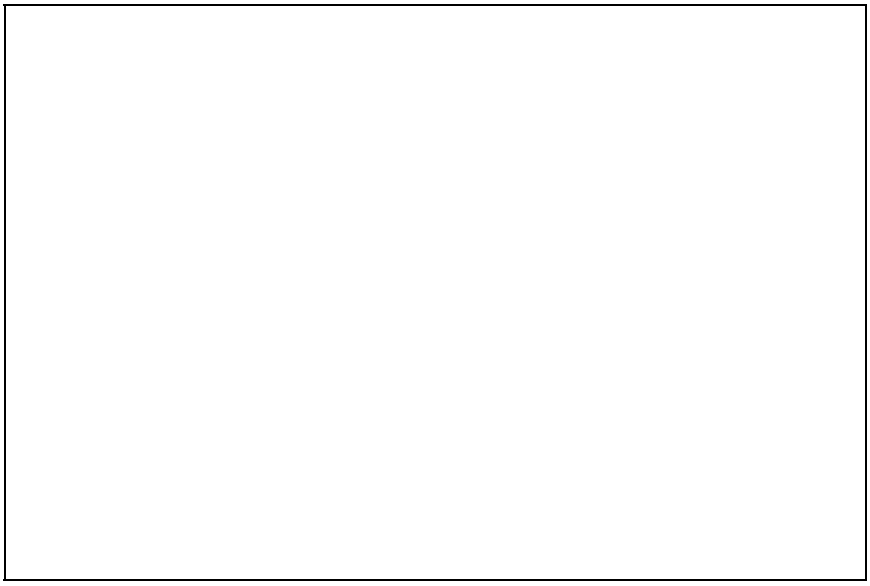


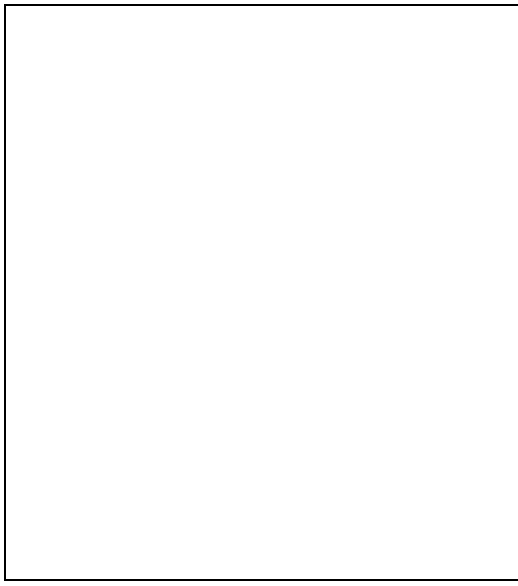
Reads/bob: two-node/1MiB





Reads/minio: two-node/1MiB





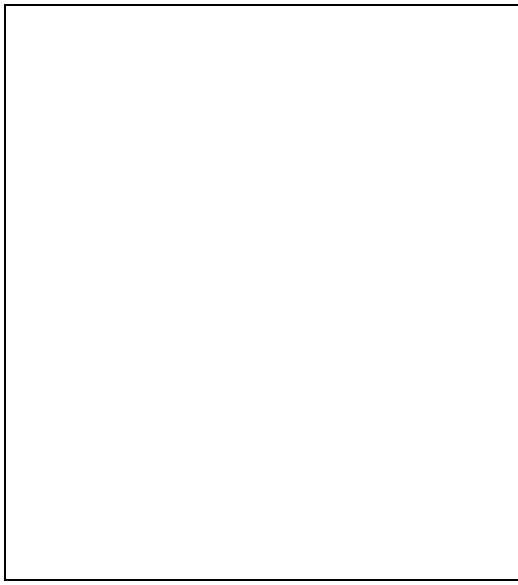
This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Reads/50KiB

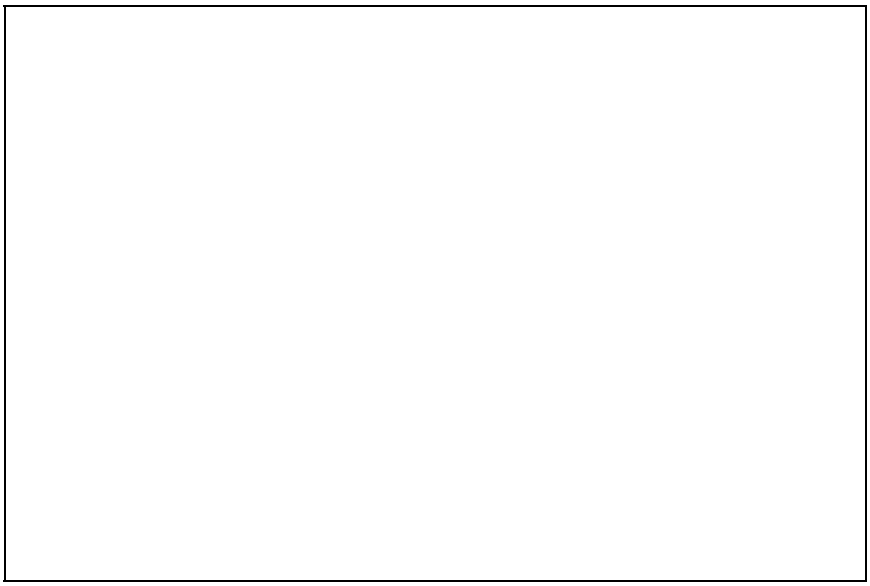
Violin Plot

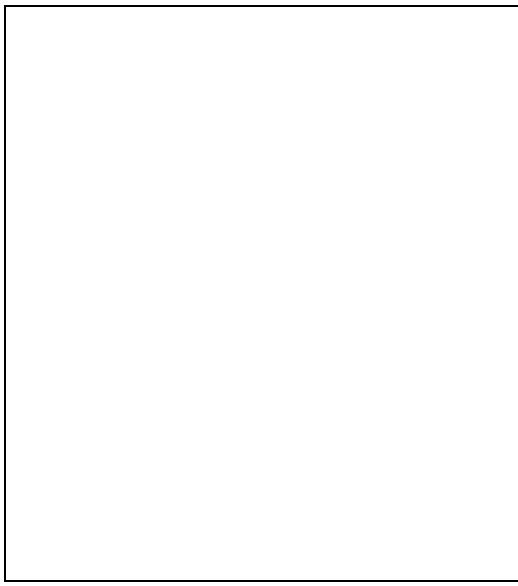
This chart shows the relationship between function/parameter and iteration time. The thickness of the shaded region indicates the probability that a measurement of the given function/parameter would take a particular length of time.

Reads/bob: one-node/50KiB

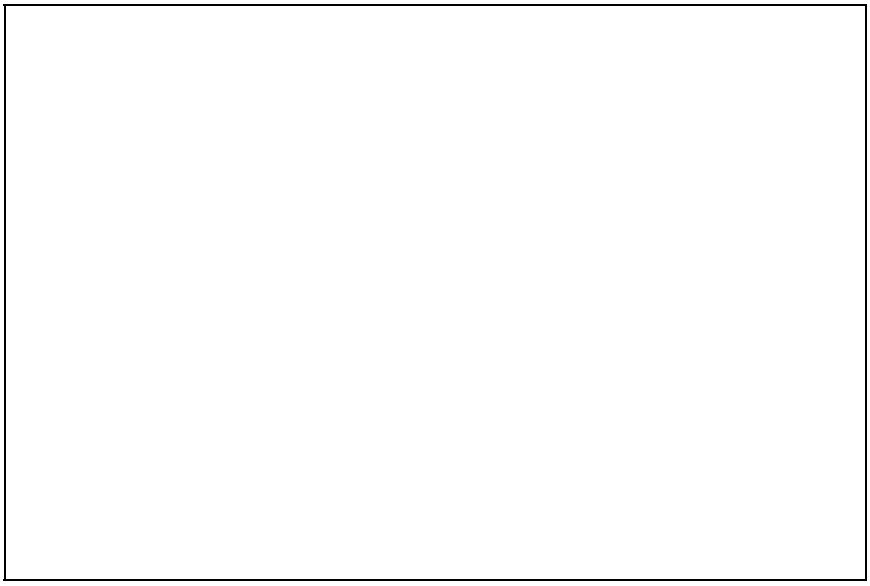


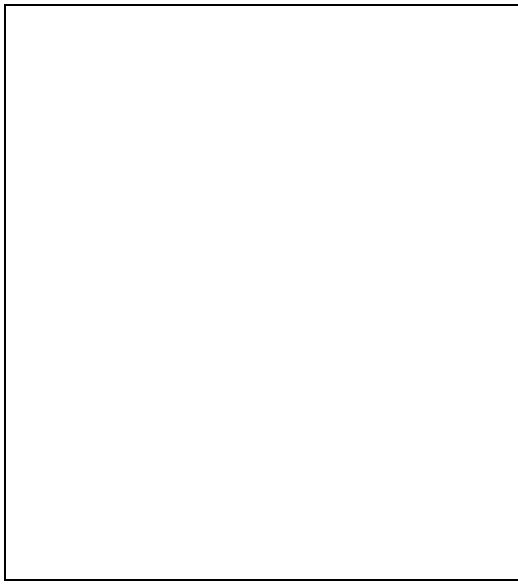
Reads/minio: one-node/50KiB



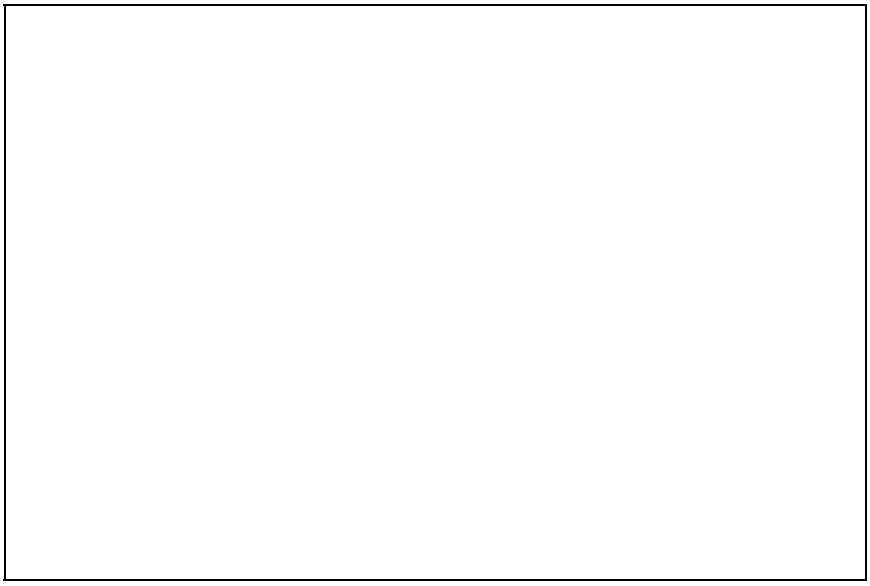


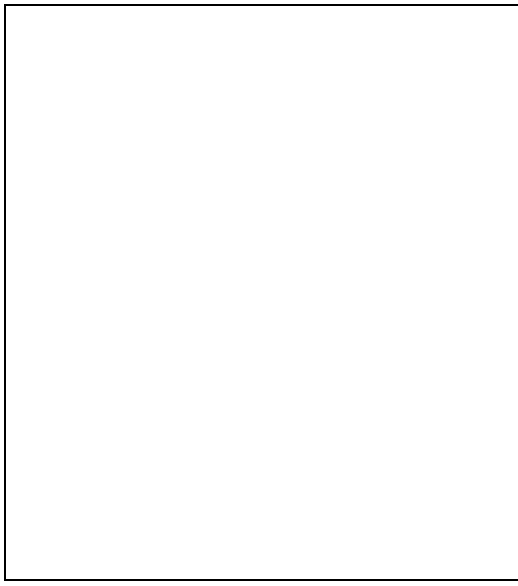
Reads/bob: two-node/50KiB





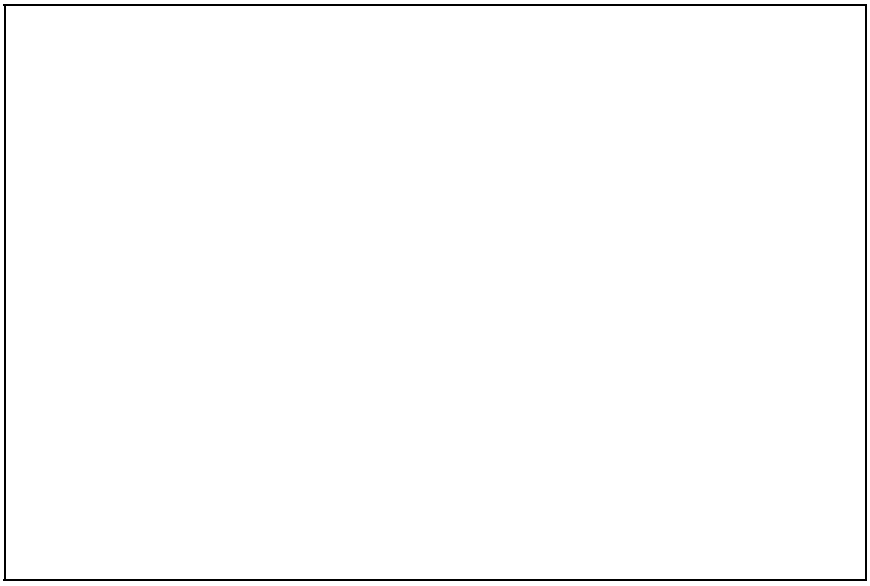
Reads/minio: two-node/50KiB

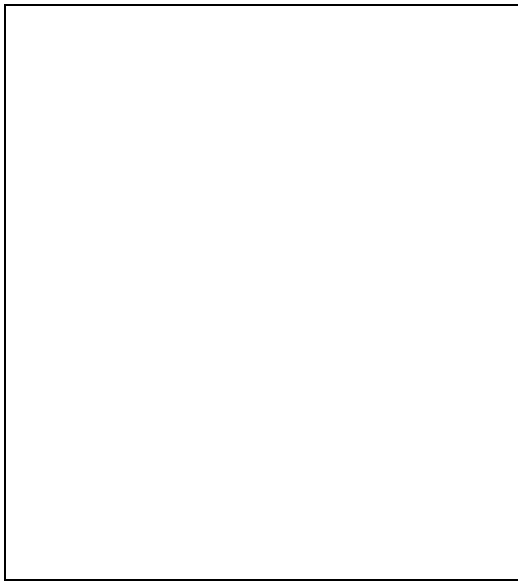




This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Reads/bob: one-node/100MiB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Throughput	96.108 MiB/s	96.837 MiB/s	97.554 MiB/s
R^2	0.0029900	0.0031014	0.0029828
Mean	1.0251 s	1.0327 s	1.0405 s
Std. Dev.	31.740 ms	39.448 ms	46.474 ms
Median	1.0258 s	1.0321 s	1.0366 s
MAD	23.686 ms	33.785 ms	42.841 ms

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD

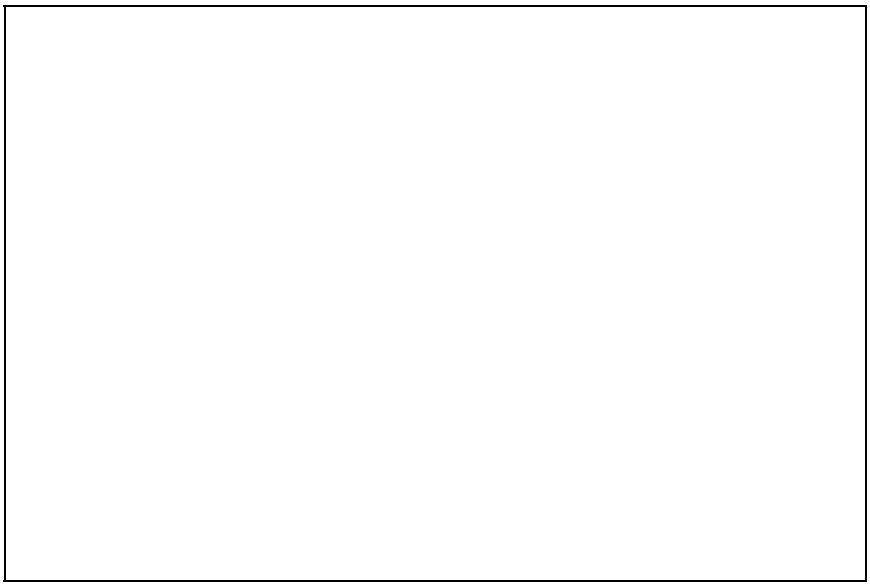
Understanding this report:

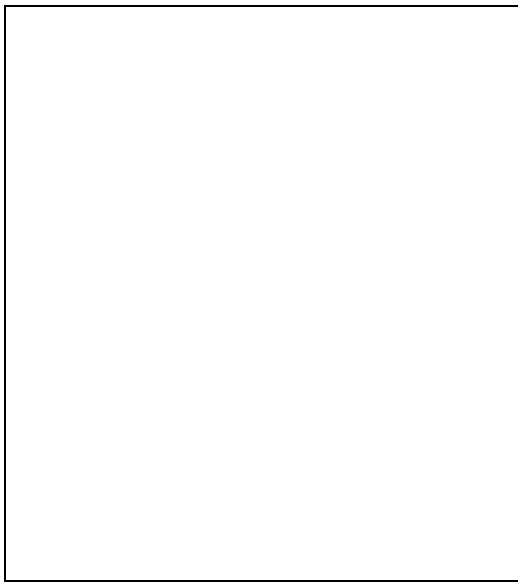
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the average time per iteration for the samples. Each point represents one sample.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	-2.5748%	-1.4918%	-0.3826%	(p = 0.01 < 0.05)
Change in throughput	+2.6429%	+1.5144%	+0.3841%	

Change within noise threshold.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

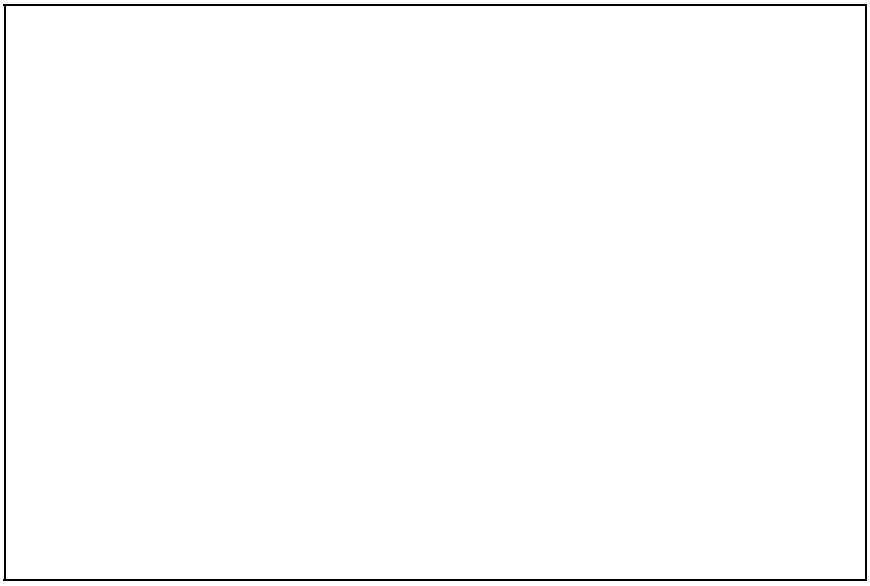
The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

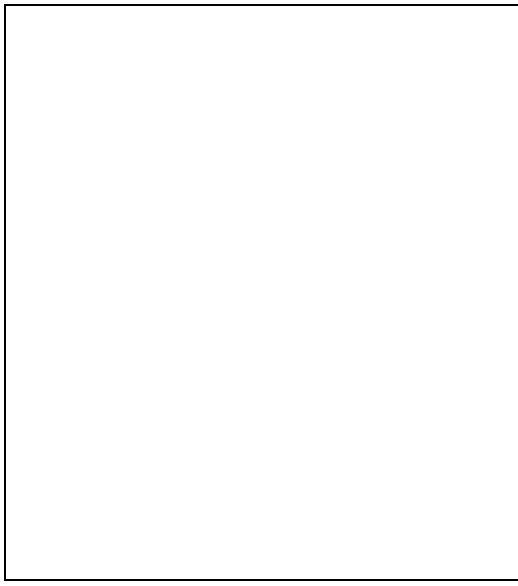
The plot on the right shows the iteration times for the two measurements. Again, the red dots represent the previous measurement while the blue dots show the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Reads/bob: one-node/1KiB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Slope	515.58 Âµs	527.50 Âµs	539.32 Âµs
Throughput	1.8107 MiB/s	1.8513 MiB/s	1.8941 MiB/s
R²	0.5308310	0.5493018	0.5311574
Mean	511.10 Âµs	520.35 Âµs	530.07 Âµs
Std. Dev.	40.218 Âµs	48.652 Âµs	56.708 Âµs
Median	495.01 Âµs	520.45 Âµs	530.90 Âµs
MAD	37.100 Âµs	50.235 Âµs	68.701 Âµs

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD
- Slope

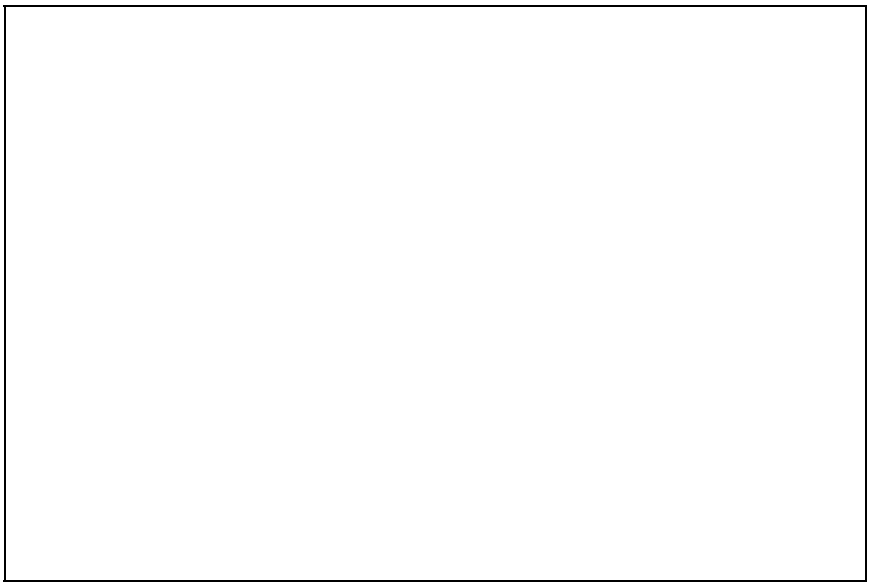
Understanding this report:

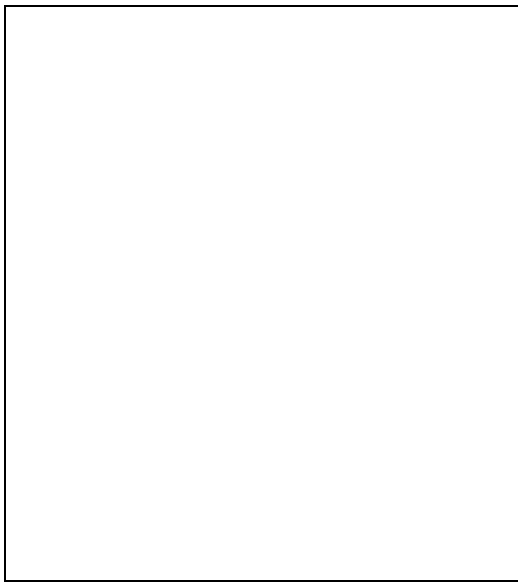
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the linear regression calculated from the measurements. Each point represents a sample, though here it shows the total time for the sample rather than time per iteration. The line is the line of best fit for these measurements.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	-3.8752%	-1.2441%	+1.2787%	(p = 0.36 > 0.05)
Change in throughput	+4.0314%	+1.2598%	-1.2625%	

No change in performance detected.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

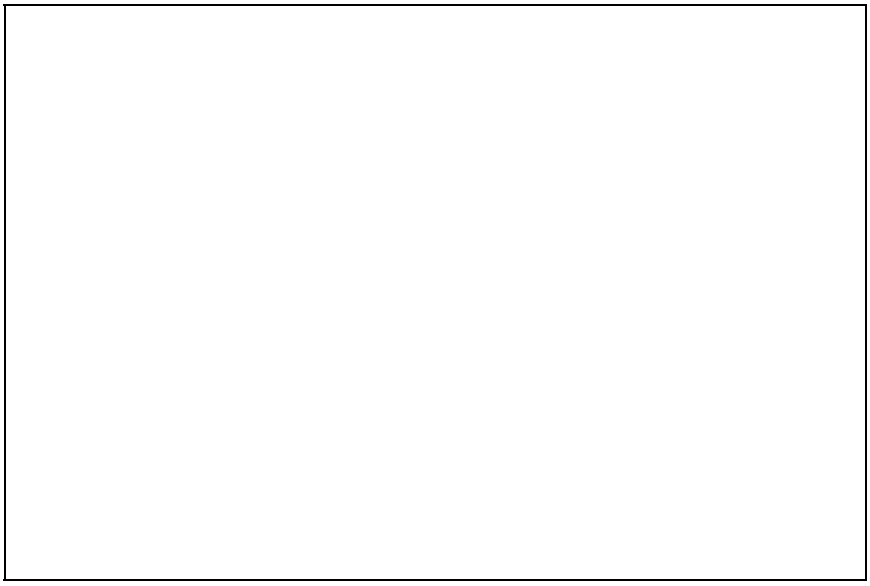
The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

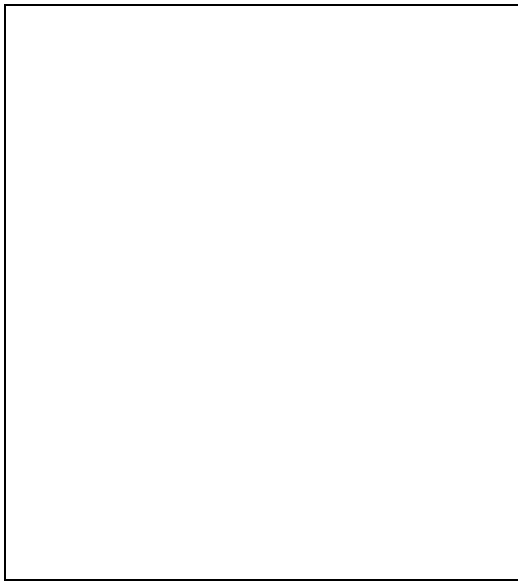
The plot on the right shows the two regressions. Again, the red line represents the previous measurement while the blue line shows the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Reads/bob: one-node/1MiB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Throughput	120.71 MiB/s	122.34 MiB/s	124.00 MiB/s
R²	0.0034234	0.0035514	0.0034205
Mean	8.0643 ms	8.1738 ms	8.2846 ms
Std. Dev.	490.62 Âµs	568.20 Âµs	639.01 Âµs
Median	8.0056 ms	8.1064 ms	8.2281 ms
MAD	405.69 Âµs	619.08 Âµs	776.27 Âµs

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD

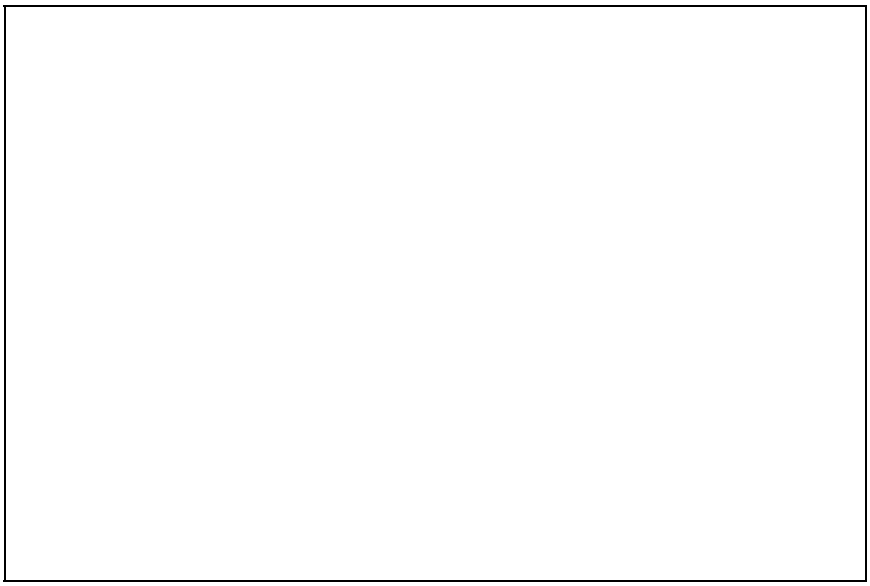
Understanding this report:

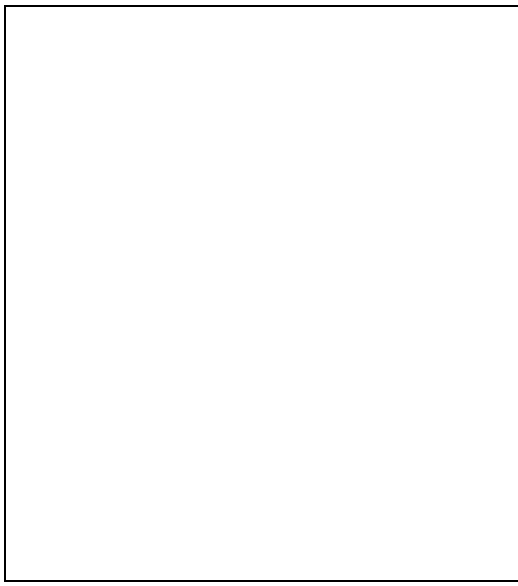
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the average time per iteration for the samples. Each point represents one sample.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	-2.4294%	-0.3233%	+1.6046%	(p = 0.75 > 0.05)
Change in throughput	+2.4899%	+0.3244%	-1.5793%	

No change in performance detected.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

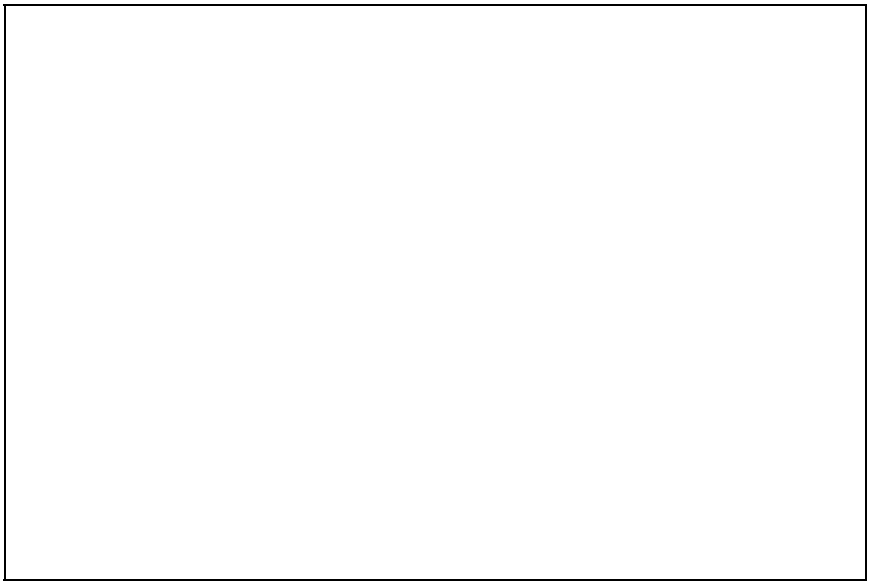
The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

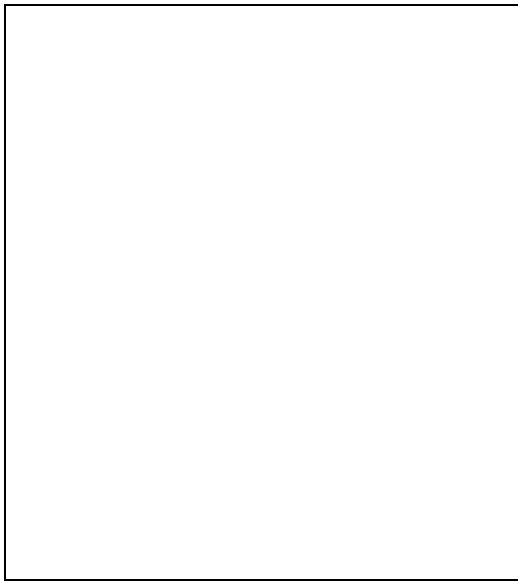
The plot on the right shows the iteration times for the two measurements. Again, the red dots represent the previous measurement while the blue dots show the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Reads/bob: one-node/50KiB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Slope	785.83 Âµs	796.88 Âµs	808.82 Âµs
Throughput	60.370 MiB/s	61.274 MiB/s	62.135 MiB/s
R²	0.6263853	0.6391565	0.6242783
Mean	776.10 Âµs	785.12 Âµs	794.50 Âµs
Std. Dev.	38.613 Âµs	47.320 Âµs	55.036 Âµs
Median	772.92 Âµs	777.41 Âµs	784.26 Âµs
MAD	20.487 Âµs	33.094 Âµs	51.438 Âµs

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD
- Slope

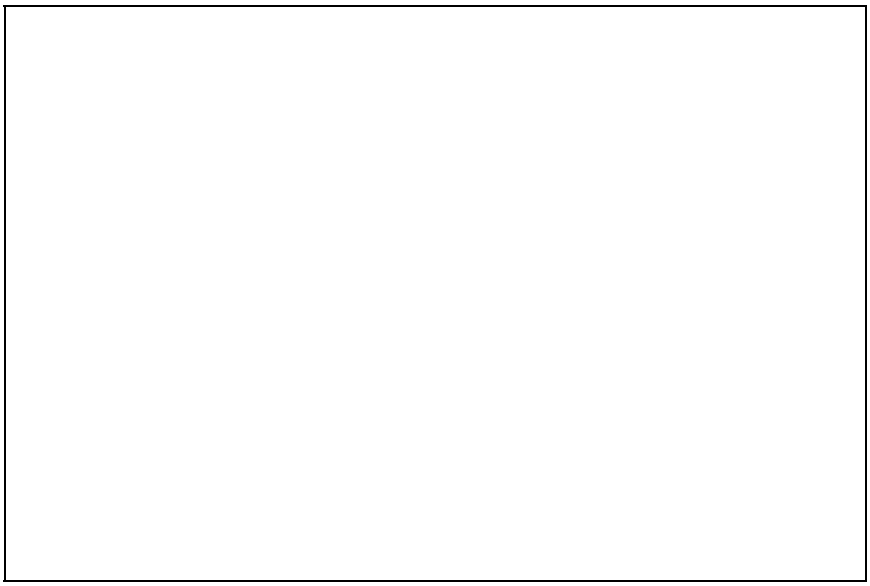
Understanding this report:

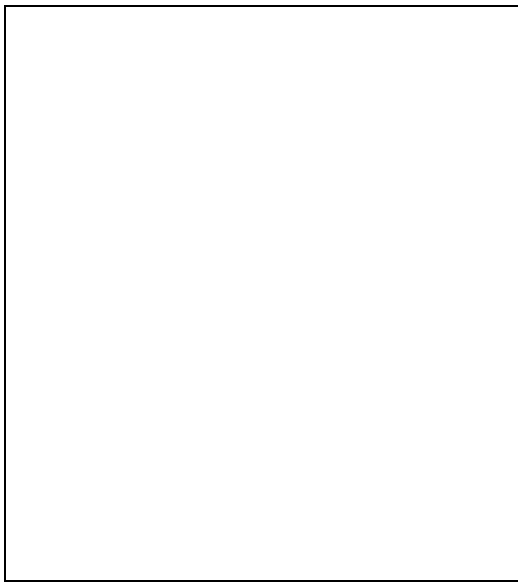
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the linear regression calculated from the measurements. Each point represents a sample, though here it shows the total time for the sample rather than time per iteration. The line is the line of best fit for these measurements.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	-0.3225%	+0.9864%	+2.3595%	(p = 0.15 > 0.05)
Change in throughput	+0.3235%	-0.9767%	-2.3051%	

No change in performance detected.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

The plot on the right shows the two regressions. Again, the red line represents the previous measurement while the blue line shows the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Reads/bob: one-node

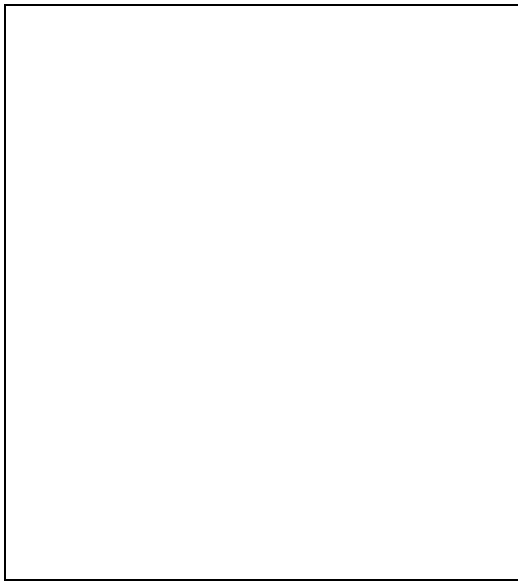
Violin Plot

This chart shows the relationship between function/parameter and iteration time. The thickness of the shaded region indicates the probability that a measurement of the given function/parameter would take a particular length of time.

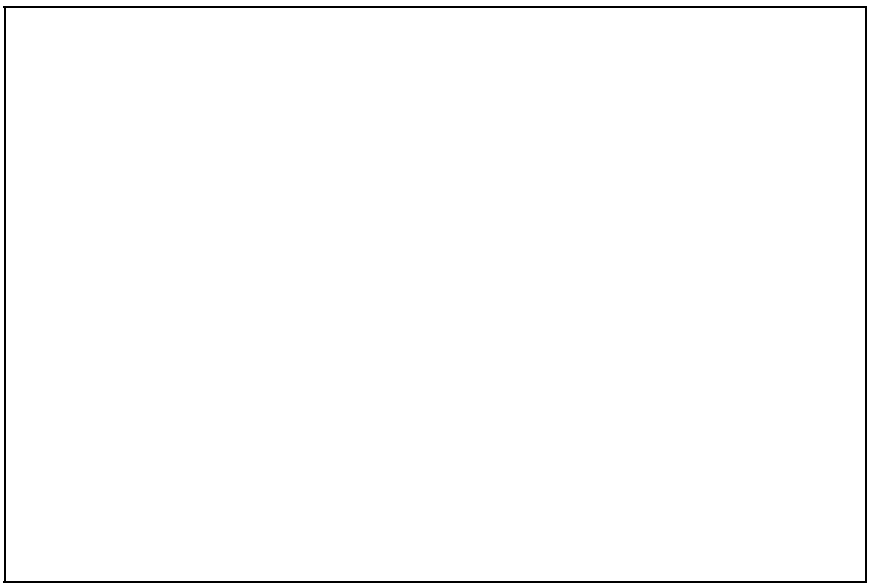
Line Chart

This chart shows the mean measured time for each function as the input (or the size of the input) increases.

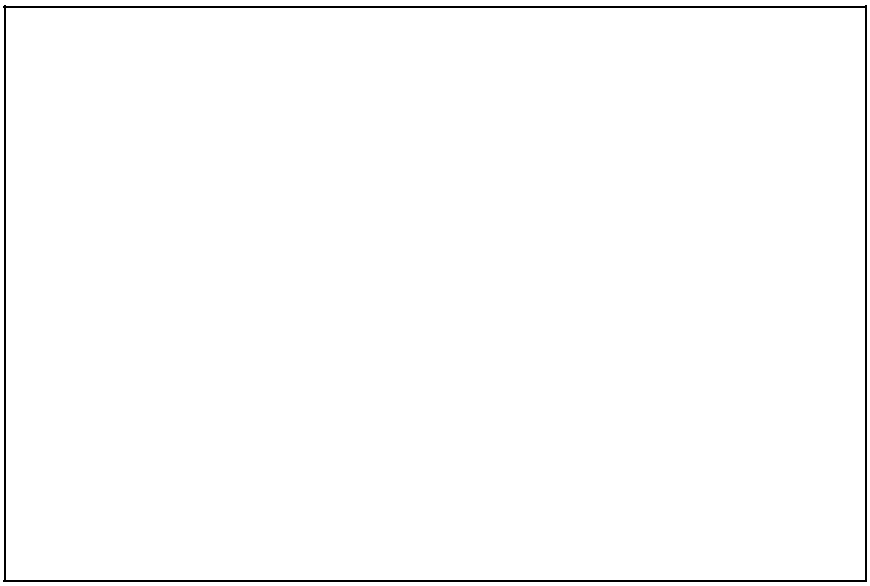
Reads/bob: one-node/1KiB



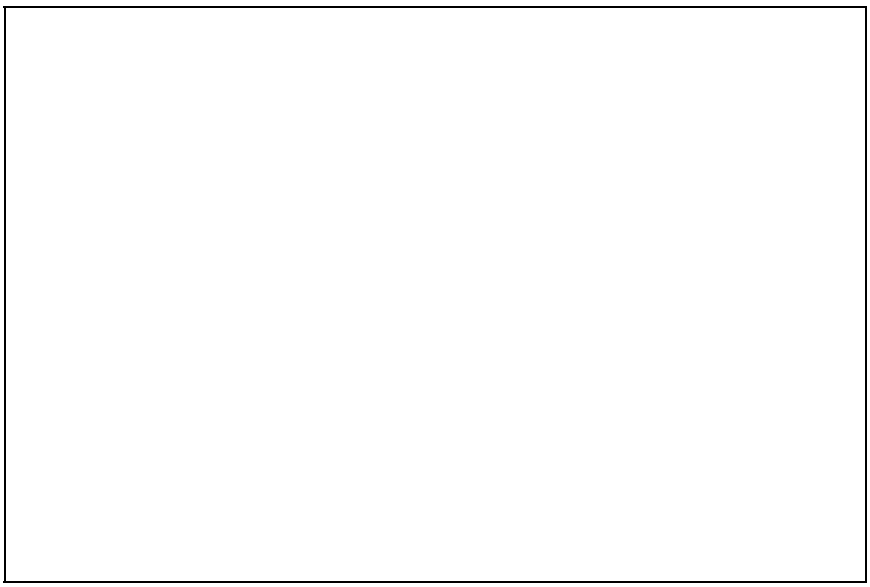
Reads/bob: one-node/50KiB

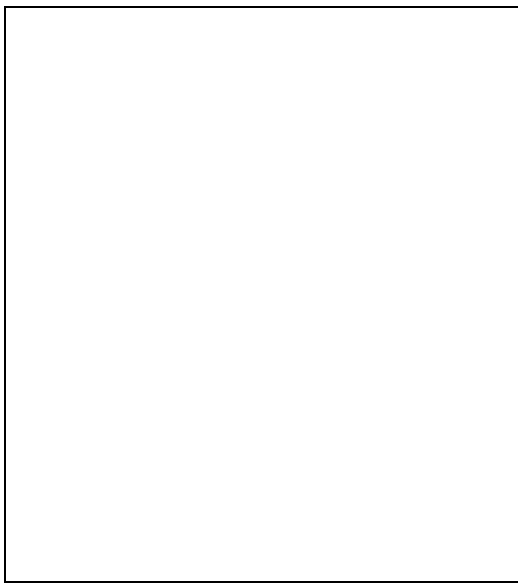


Reads/bob: one-node/1MiB



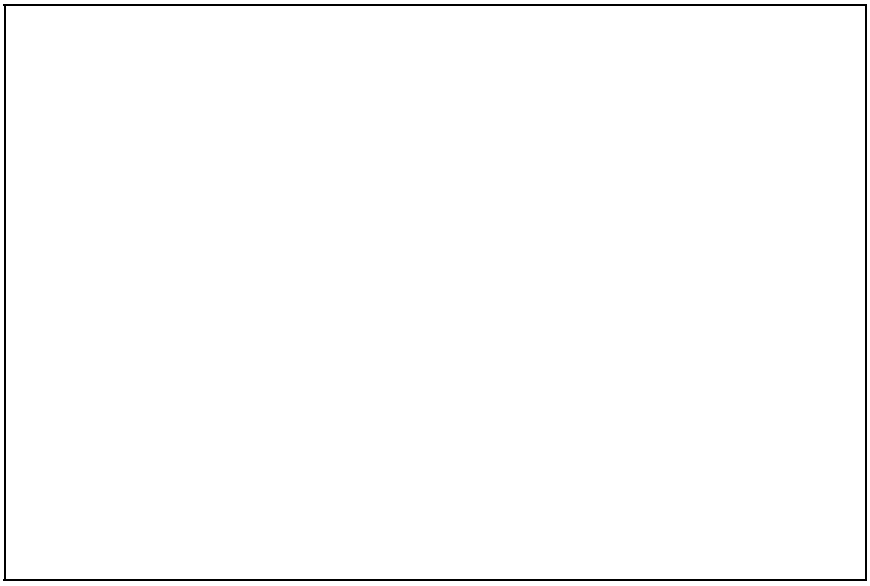
Reads/bob: one-node/100MiB

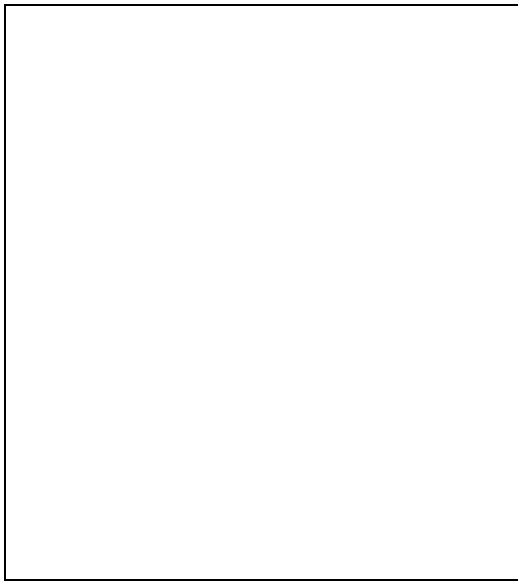




This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Reads/bob: two-node/100MiB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Throughput	94.372 MiB/s	95.076 MiB/s	95.815 MiB/s
R^2	0.0006404	0.0006656	0.0006419
Mean	1.0437 s	1.0518 s	1.0596 s
Std. Dev.	34.174 ms	41.044 ms	47.431 ms
Median	1.0548 s	1.0599 s	1.0668 s
MAD	21.577 ms	33.845 ms	45.426 ms

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD

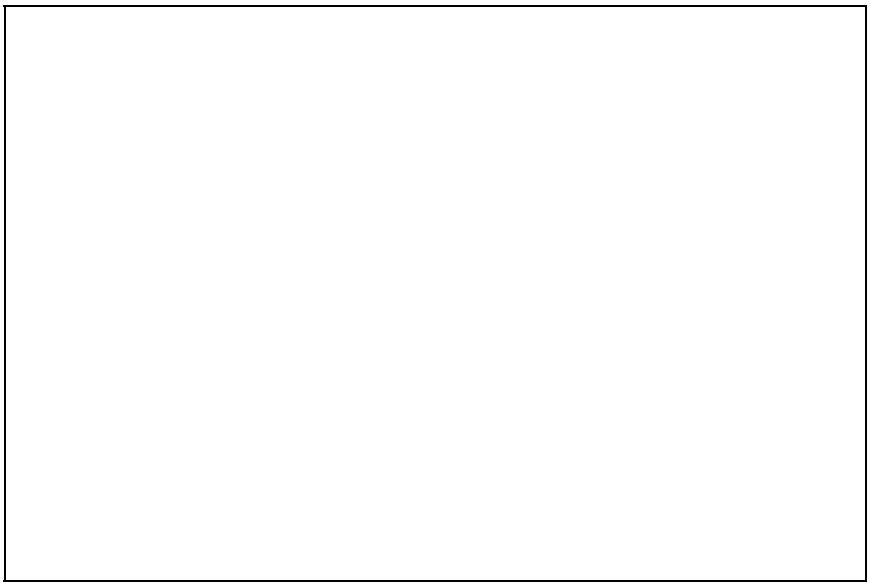
Understanding this report:

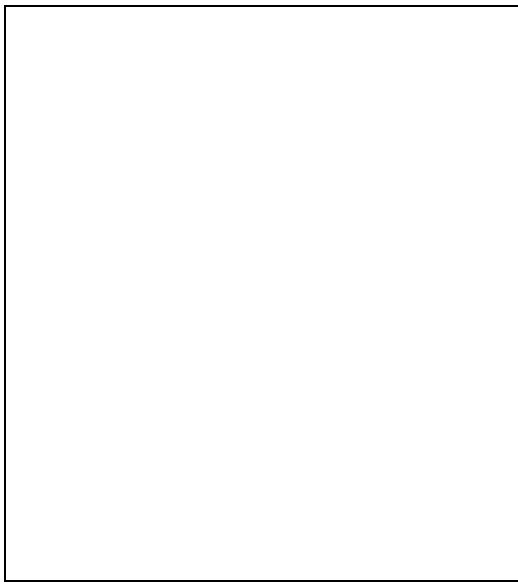
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the average time per iteration for the samples. Each point represents one sample.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	-1.3506%	-0.1925%	+0.9386%	(p = 0.75 > 0.05)
Change in throughput	+1.3691%	+0.1929%	-0.9299%	

No change in performance detected.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

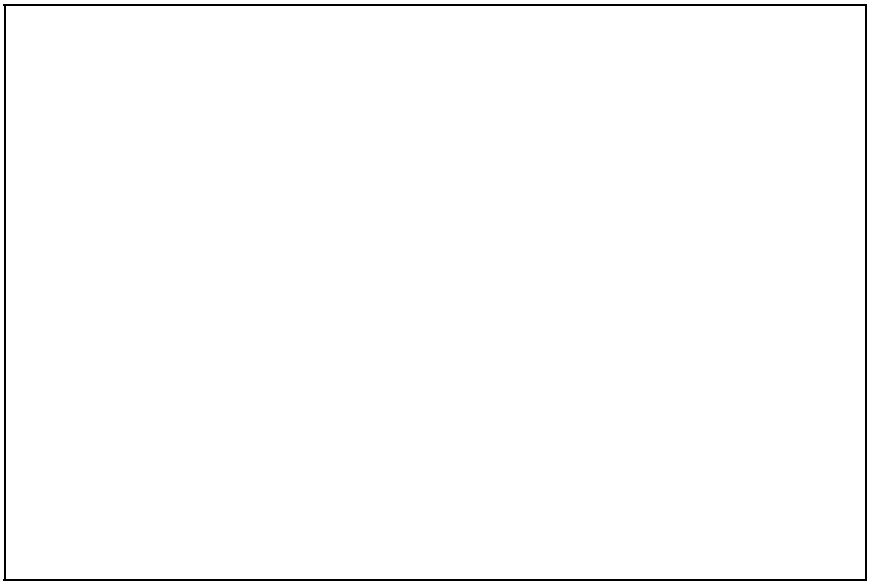
The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

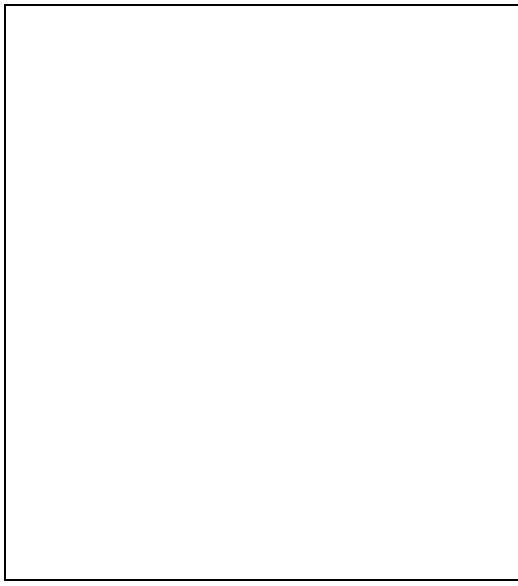
The plot on the right shows the iteration times for the two measurements. Again, the red dots represent the previous measurement while the blue dots show the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Reads/bob: two-node/1KiB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Slope	243.18 Âµs	245.55 Âµs	248.08 Âµs
Throughput	3.9364 MiB/s	3.9770 MiB/s	4.0158 MiB/s
R²	0.7948653	0.8042353	0.7935401
Mean	240.18 Âµs	242.24 Âµs	244.39 Âµs
Std. Dev.	8.9940 Âµs	10.792 Âµs	12.384 Âµs
Median	236.65 Âµs	239.32 Âµs	242.95 Âµs
MAD	7.0904 Âµs	9.2224 Âµs	12.917 Âµs

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD
- Slope

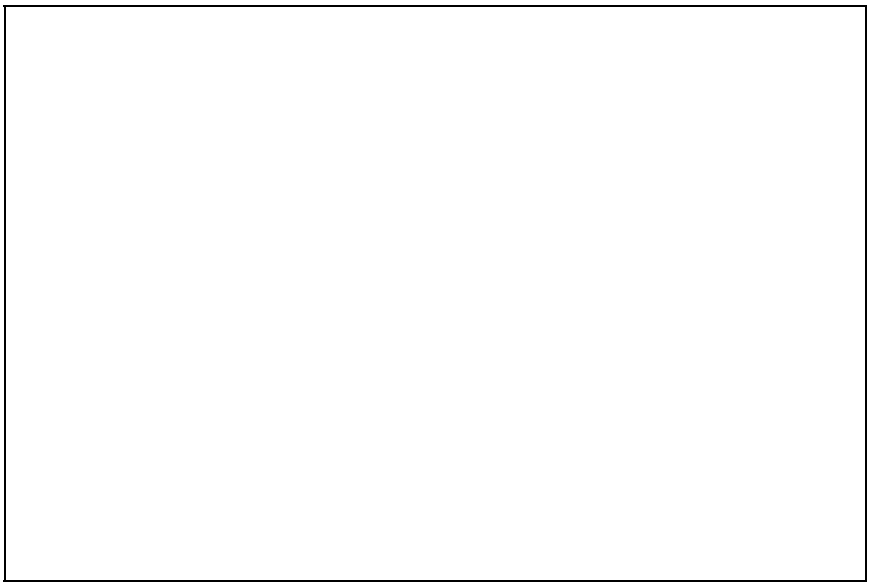
Understanding this report:

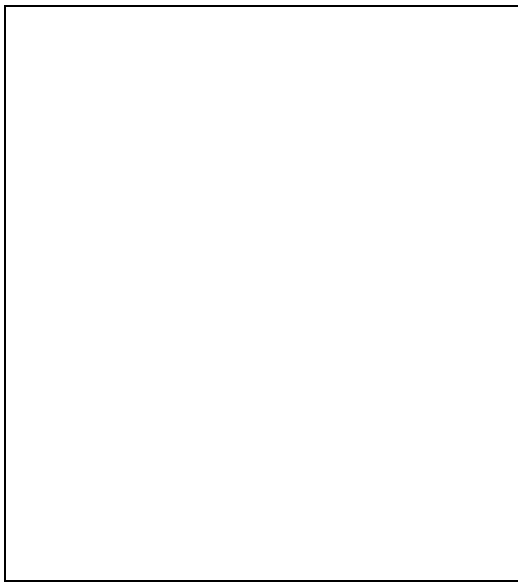
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the linear regression calculated from the measurements. Each point represents a sample, though here it shows the total time for the sample rather than time per iteration. The line is the line of best fit for these measurements.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	-2.0561%	-0.8270%	+0.3965%	(p = 0.20 > 0.05)
Change in throughput	+2.0993%	+0.8339%	-0.3949%	

No change in performance detected.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

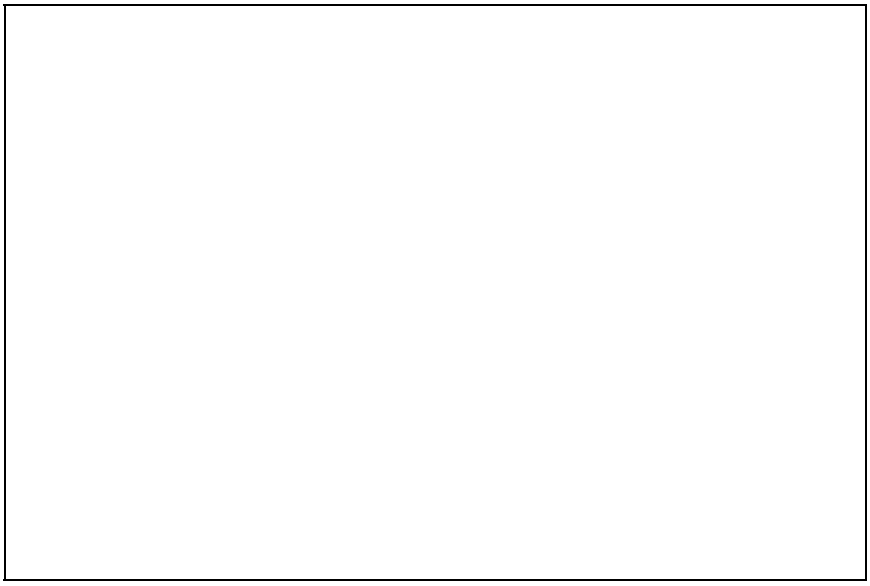
The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

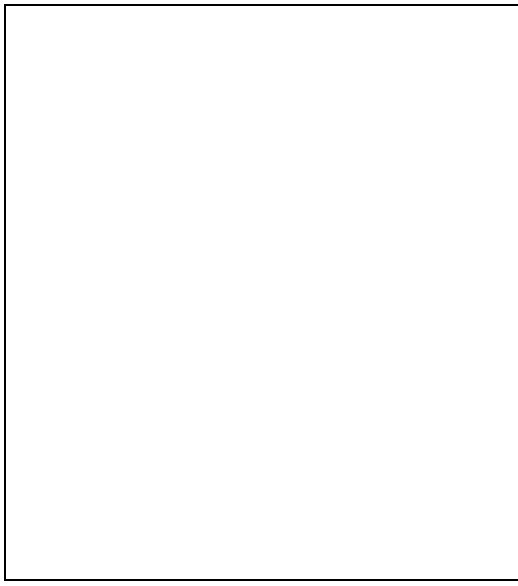
The plot on the right shows the two regressions. Again, the red line represents the previous measurement while the blue line shows the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Reads/bob: two-node/1MiB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Throughput	113.29 MiB/s	115.13 MiB/s	116.99 MiB/s
R²	0.0013160	0.0013653	0.0013137
Mean	8.5479 ms	8.6859 ms	8.8273 ms
Std. Dev.	588.98 Âµs	715.97 Âµs	842.40 Âµs
Median	8.5075 ms	8.7017 ms	8.8593 ms
MAD	552.98 Âµs	700.38 Âµs	863.32 Âµs

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD

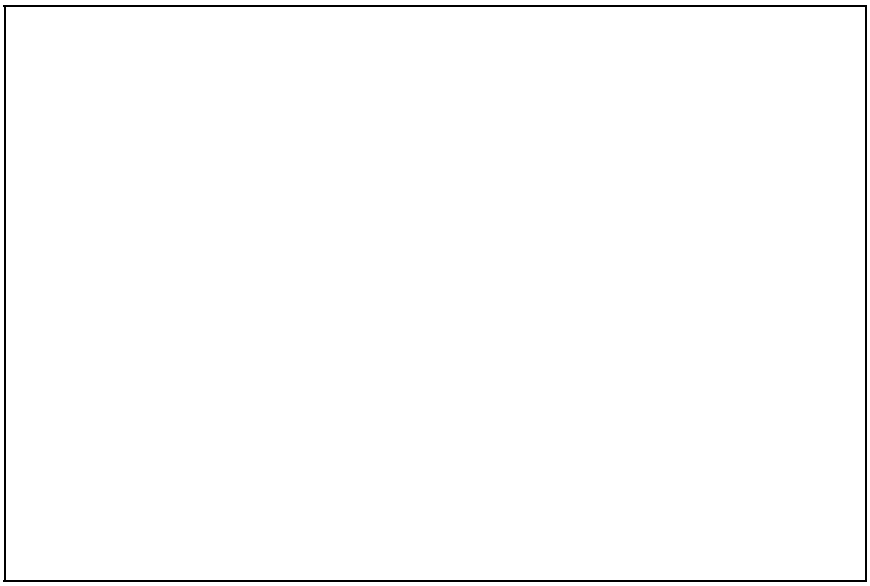
Understanding this report:

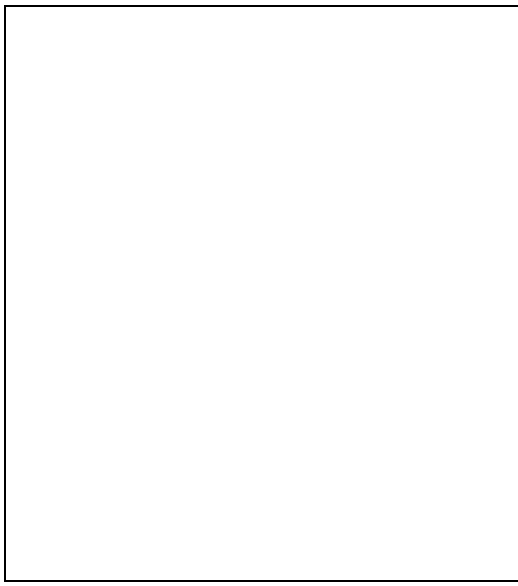
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the average time per iteration for the samples. Each point represents one sample.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	-2.9632%	-0.8705%	+1.2668%	(p = 0.43 > 0.05)
Change in throughput	+3.0537%	+0.8782%	-1.2509%	

No change in performance detected.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

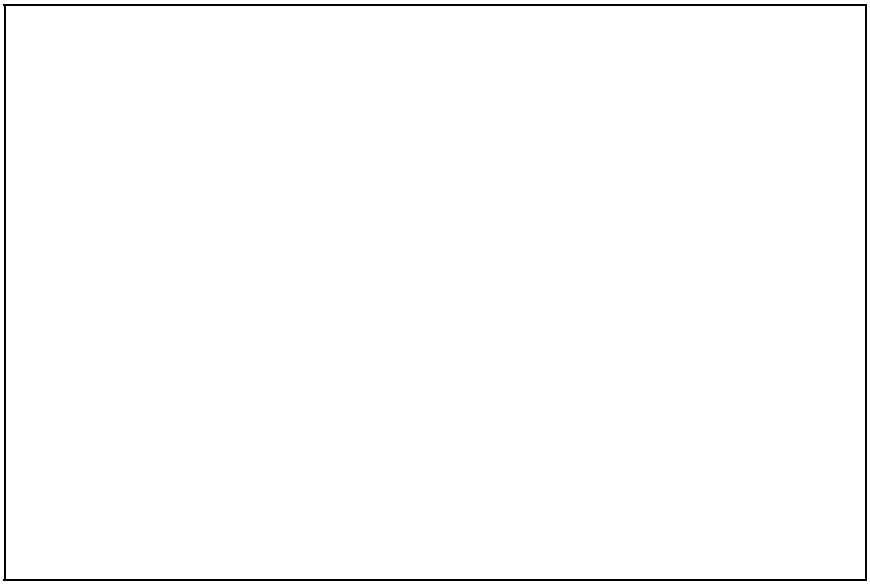
The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

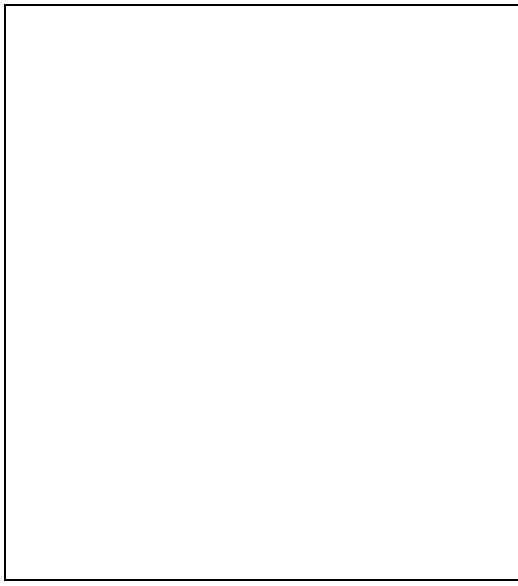
The plot on the right shows the iteration times for the two measurements. Again, the red dots represent the previous measurement while the blue dots show the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Reads/bob: two-node/50KiB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Slope	603.60 Âµs	610.18 Âµs	617.61 Âµs
Throughput	79.059 MiB/s	80.022 MiB/s	80.895 MiB/s
R²	0.6556132	0.6636919	0.6534411
Mean	603.10 Âµs	612.58 Âµs	626.01 Âµs
Std. Dev.	23.444 Âµs	59.878 Âµs	95.587 Âµs
Median	603.37 Âµs	607.40 Âµs	611.51 Âµs
MAD	13.623 Âµs	17.606 Âµs	21.731 Âµs

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD
- Slope

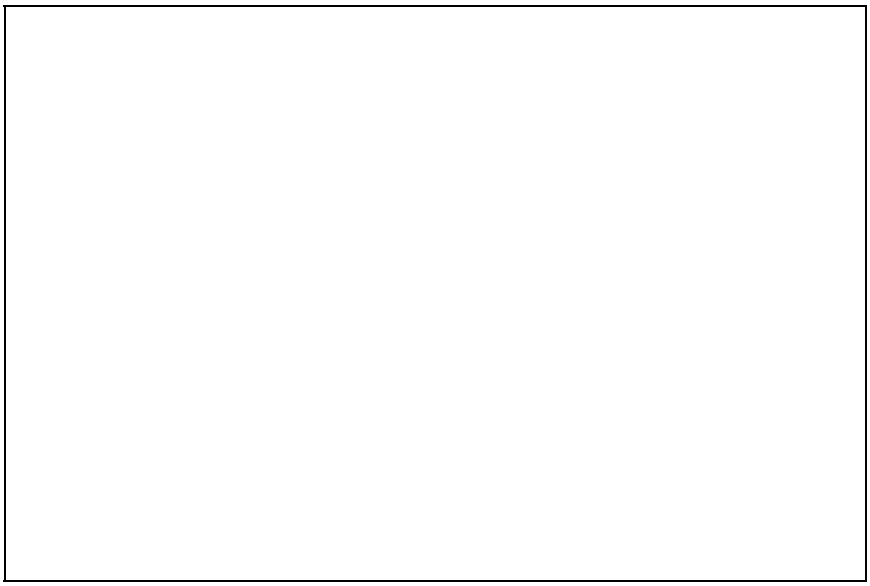
Understanding this report:

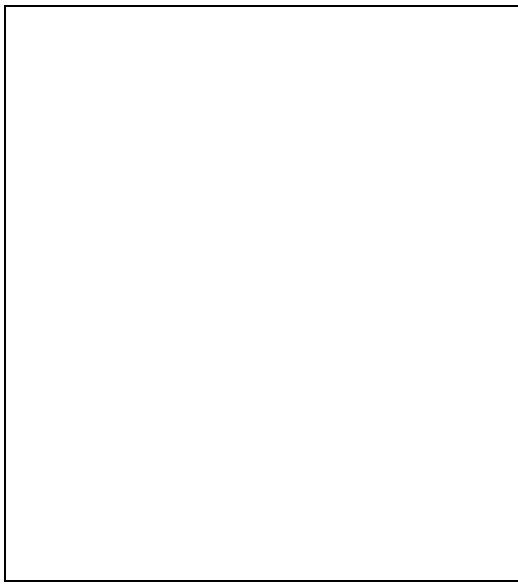
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the linear regression calculated from the measurements. Each point represents a sample, though here it shows the total time for the sample rather than time per iteration. The line is the line of best fit for these measurements.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	+0.2726%	+2.1404%	+4.5655%	(p = 0.05 < 0.05)
Change in throughput	-0.2718%	-2.0955%	-4.3662%	

Change within noise threshold.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

The plot on the right shows the two regressions. Again, the red line represents the previous measurement while the blue line shows the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Reads/bob: two-node

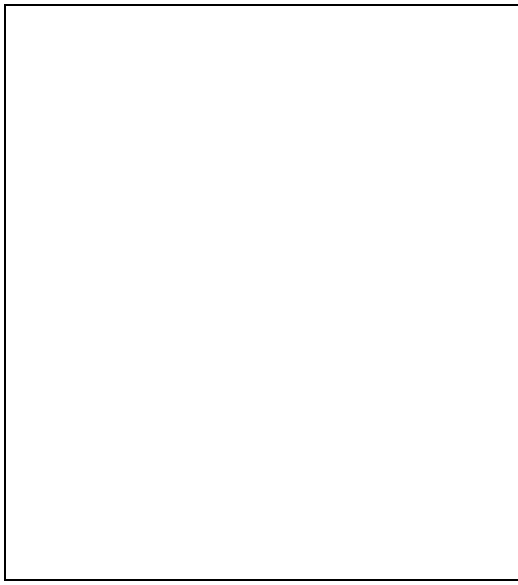
Violin Plot

This chart shows the relationship between function/parameter and iteration time. The thickness of the shaded region indicates the probability that a measurement of the given function/parameter would take a particular length of time.

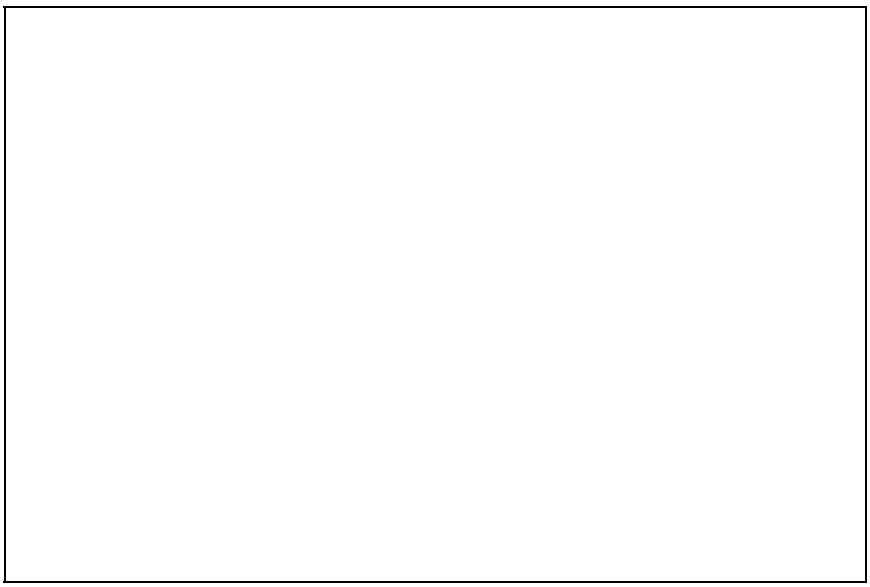
Line Chart

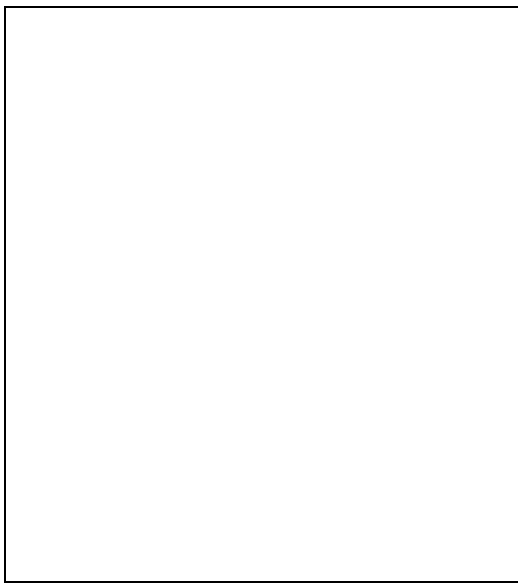
This chart shows the mean measured time for each function as the input (or the size of the input) increases.

Reads/bob: two-node/1KiB

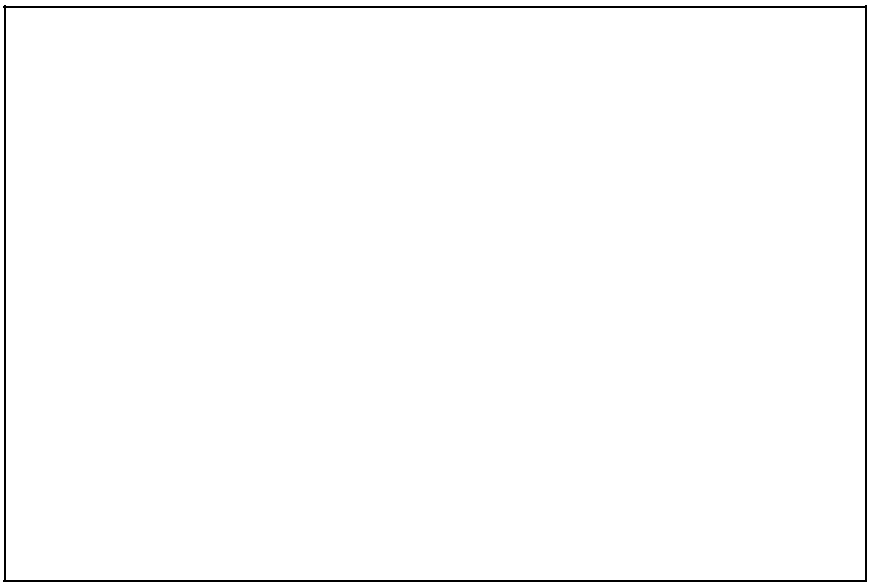


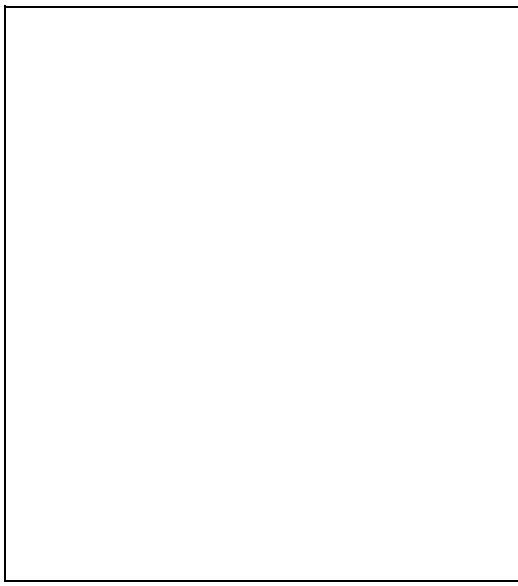
Reads/bob: two-node/50KiB



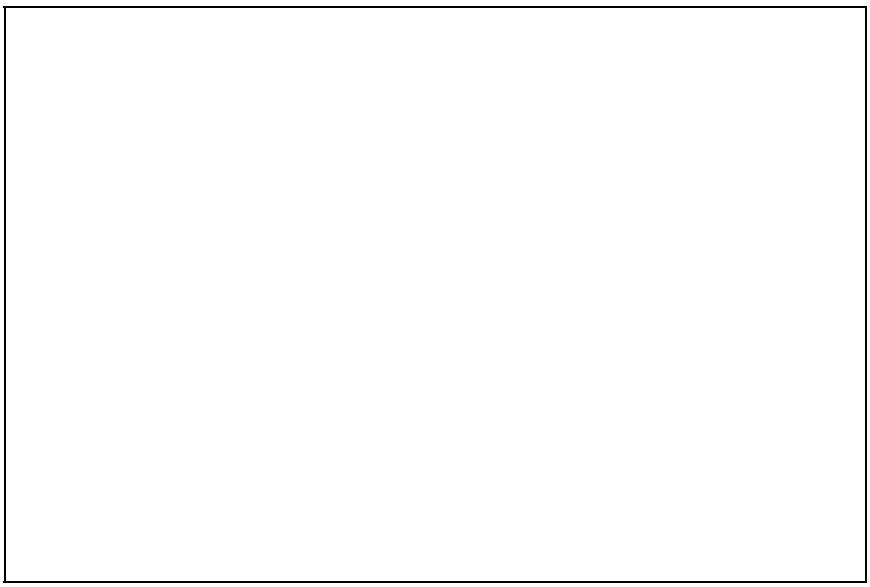


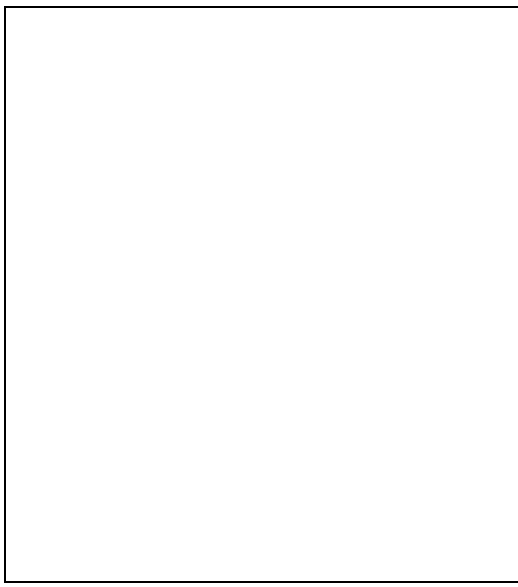
Reads/bob: two-node/1MiB





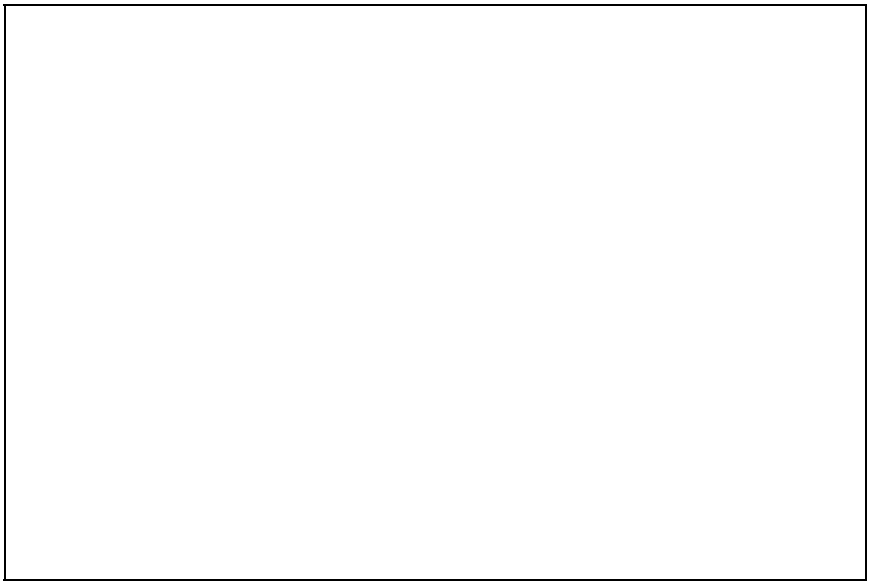
Reads/bob: two-node/100MiB

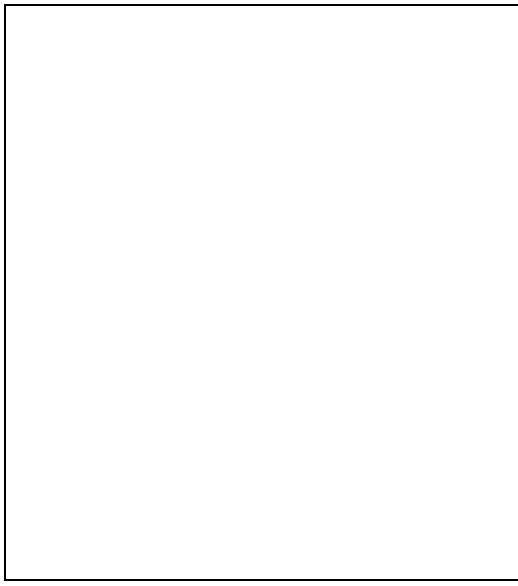




This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Reads/minio: one-node/0B





Additional Statistics:

	Lower bound	Estimate	Upper bound
Slope	1.2559 ms	1.2750 ms	1.2971 ms
Throughput	770.95 KiB/s	784.32 KiB/s	796.23 KiB/s
R²	0.5578653	0.5715106	0.5533465
Mean	1.2624 ms	1.2793 ms	1.3002 ms
Std. Dev.	52.657 Âµs	96.897 Âµs	141.72 Âµs
Median	1.2436 ms	1.2598 ms	1.2722 ms
MAD	30.647 Âµs	37.597 Âµs	50.446 Âµs

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD
- Slope

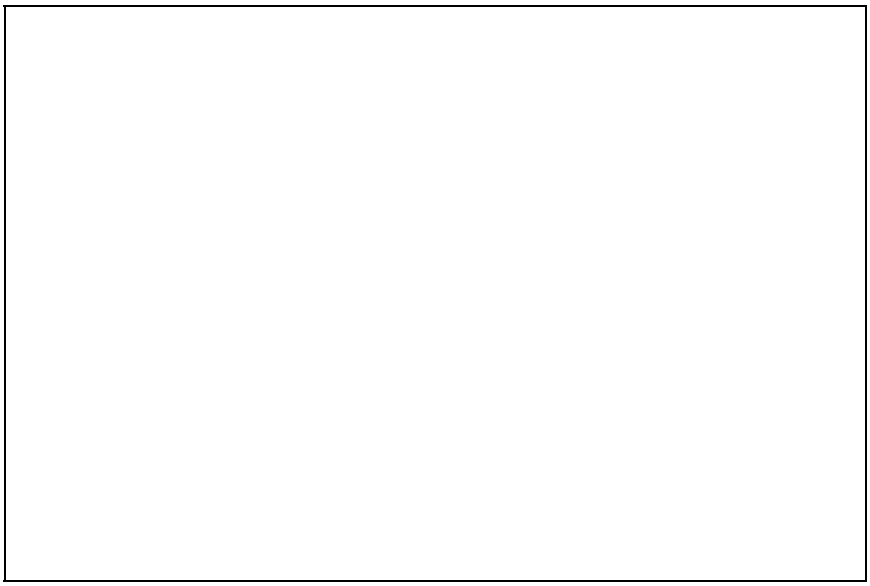
Understanding this report:

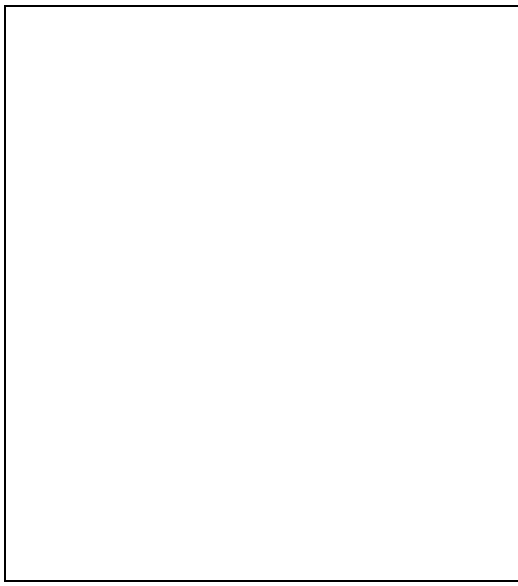
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the linear regression calculated from the measurements. Each point represents a sample, though here it shows the total time for the sample rather than time per iteration. The line is the line of best fit for these measurements.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	-61.356%	-59.648%	-57.922%	(p = 0.00 < 0.05)
Change in throughput	+158.77%	+147.82%	+137.65%	

Performance has improved.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

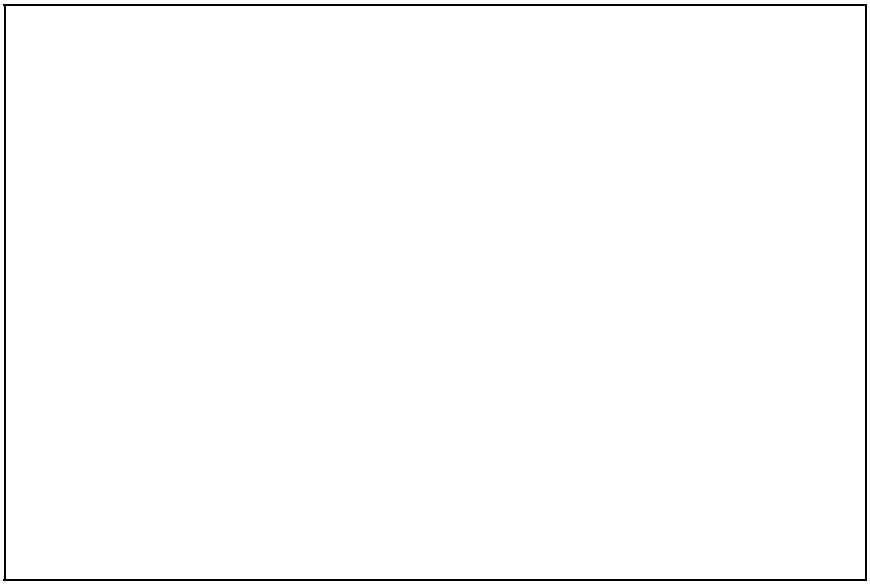
The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

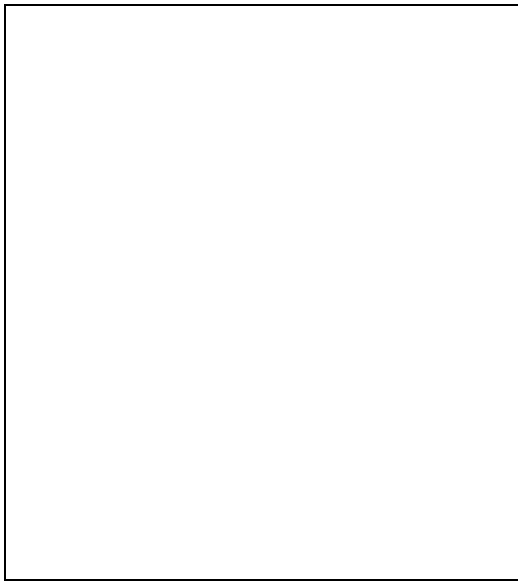
The plot on the right shows the two regressions. Again, the red line represents the previous measurement while the blue line shows the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Reads/minio: one-node/0B #2





Additional Statistics:

	Lower bound	Estimate	Upper bound
Slope	1.7393 ms	1.7782 ms	1.8248 ms
Throughput	26.758 MiB/s	27.460 MiB/s	28.074 MiB/s
R²	0.4603150	0.4737588	0.4546962
Mean	1.6637 ms	1.6969 ms	1.7332 ms
Std. Dev.	125.85 Âµs	178.02 Âµs	235.95 Âµs
Median	1.6769 ms	1.6961 ms	1.7228 ms
MAD	86.134 Âµs	126.95 Âµs	207.37 Âµs

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD
- Slope

Understanding this report:

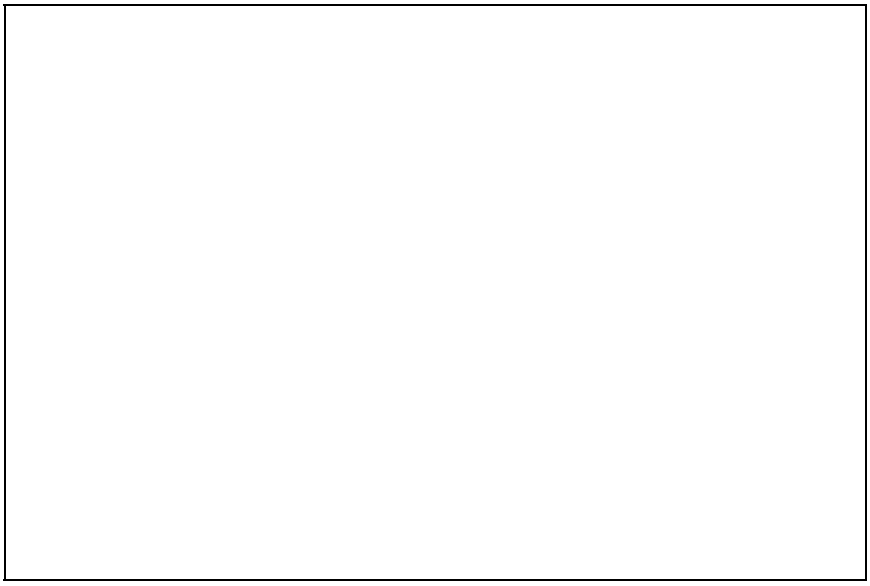
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

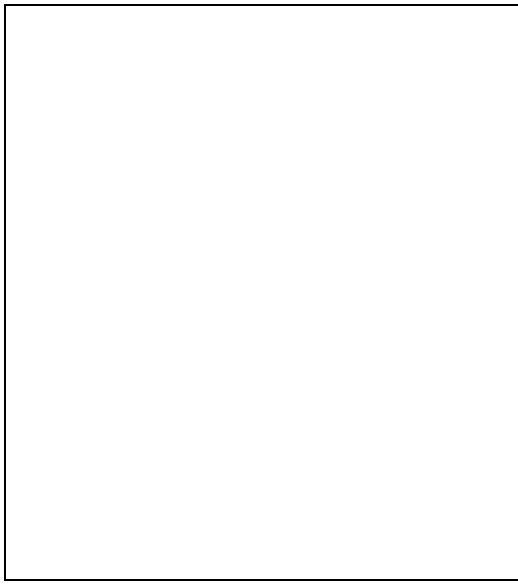
The plot on the right shows the linear regression calculated from the measurements. Each point represents a sample, though here it shows the total time for the sample rather than time per iteration. The line is the line of best fit for these measurements.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Reads/minio: one-node/0B #3





Additional Statistics:

	Lower bound	Estimate	Upper bound
Throughput	369.84 MiB/s	384.80 MiB/s	399.19 MiB/s
R²	0.0089378	0.0092415	0.0088623
Mean	2.5050 ms	2.5987 ms	2.7038 ms
Std. Dev.	347.48 Âµs	508.37 Âµs	678.79 Âµs
Median	2.4979 ms	2.5726 ms	2.6299 ms
MAD	186.80 Âµs	424.15 Âµs	511.16 Âµs

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD

Understanding this report:

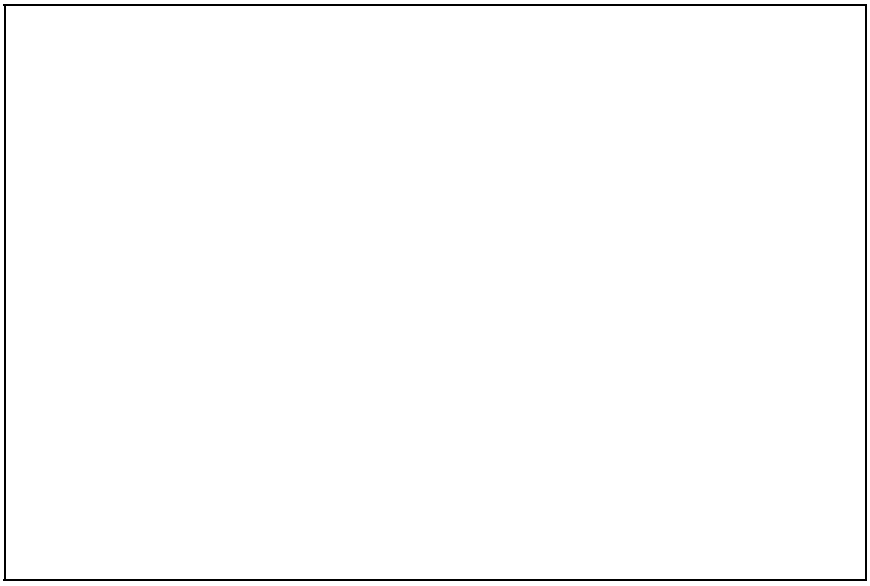
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

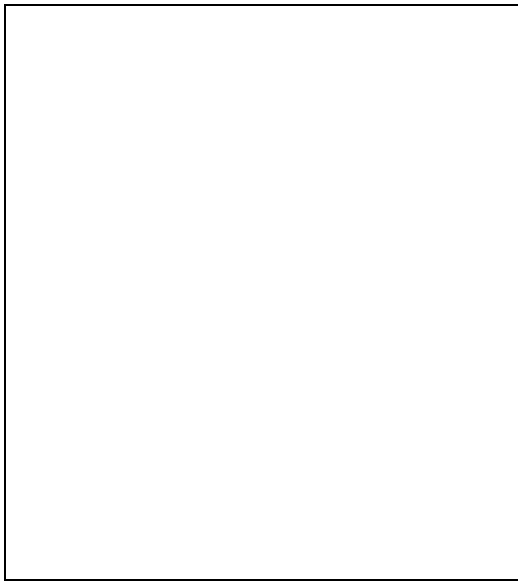
The plot on the right shows the average time per iteration for the samples. Each point represents one sample.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Reads/minio: one-node/0B #4





Additional Statistics:

	Lower bound	Estimate	Upper bound
Throughput	26.636 GiB/s	28.536 GiB/s	30.566 GiB/s
R²	0.0077774	0.0080532	0.0077362
Mean	3.1950 ms	3.4222 ms	3.6664 ms
Std. Dev.	950.93 Âµs	1.2076 ms	1.4401 ms
Median	2.7761 ms	3.0130 ms	3.4809 ms
MAD	658.51 Âµs	943.52 Âµs	1.3409 ms

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD

Understanding this report:

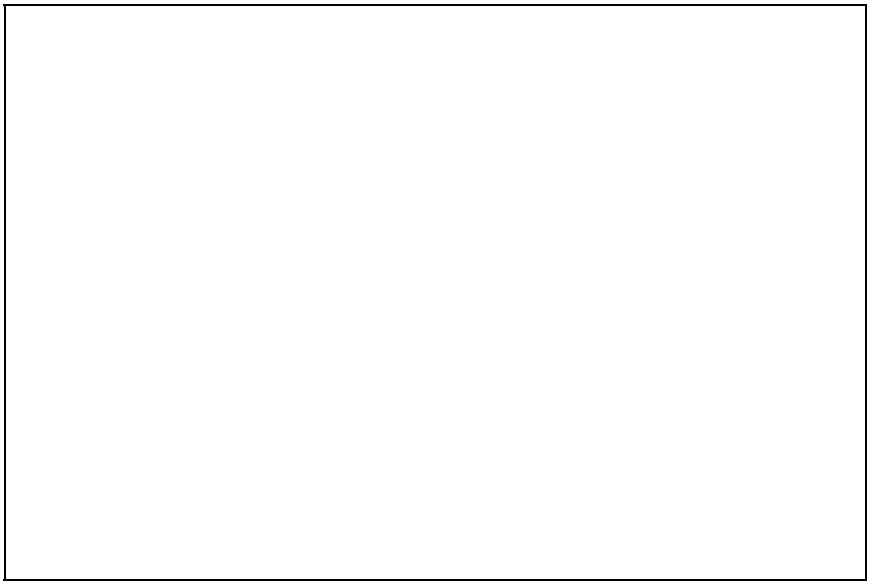
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

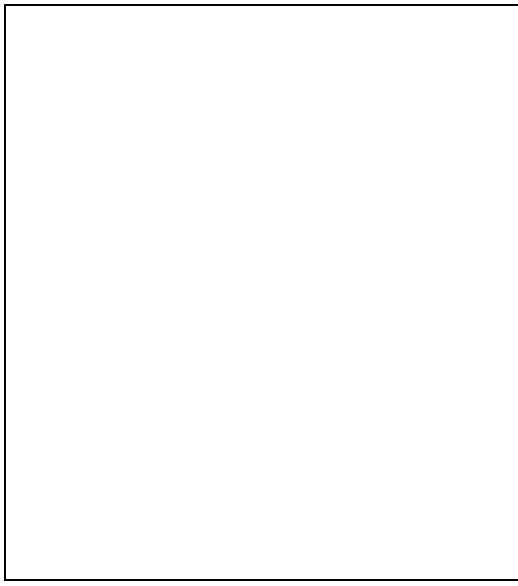
The plot on the right shows the average time per iteration for the samples. Each point represents one sample.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Reads/minio: one-node/100MiB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Throughput	208.14 MiB/s	211.21 MiB/s	213.99 MiB/s
R^2	0.0045557	0.0047090	0.0045136
Mean	467.31 ms	473.47 ms	480.46 ms
Std. Dev.	21.571 ms	33.655 ms	43.205 ms
Median	463.74 ms	468.47 ms	471.69 ms
MAD	12.759 ms	17.099 ms	21.112 ms

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD

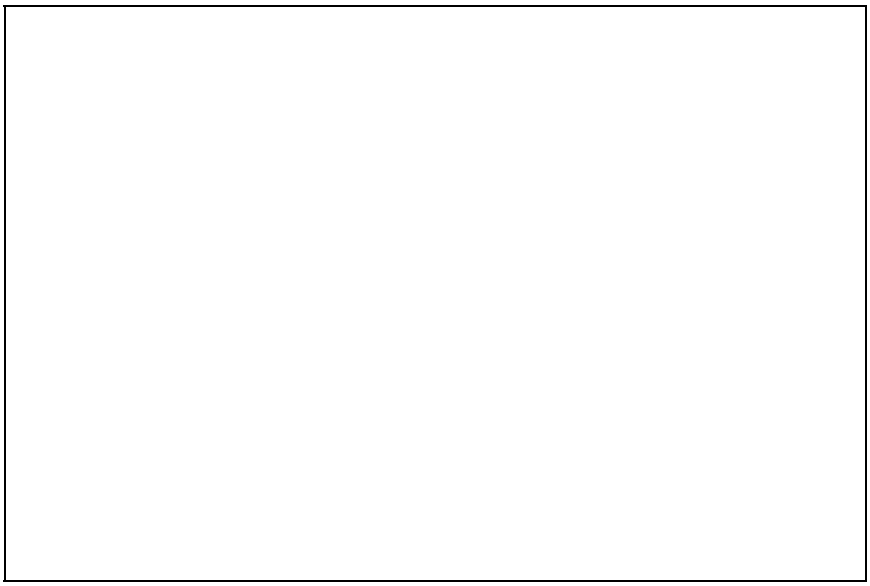
Understanding this report:

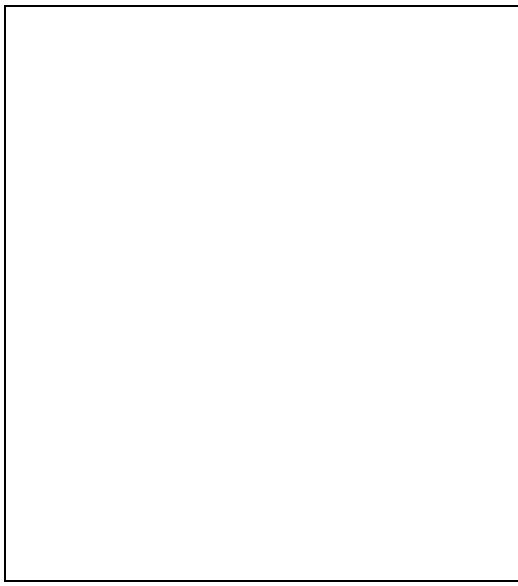
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the average time per iteration for the samples. Each point represents one sample.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	-0.6047%	+0.9443%	+2.6117%	(p = 0.26 > 0.05)
Change in throughput	+0.6083%	-0.9355%	-2.5452%	

No change in performance detected.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

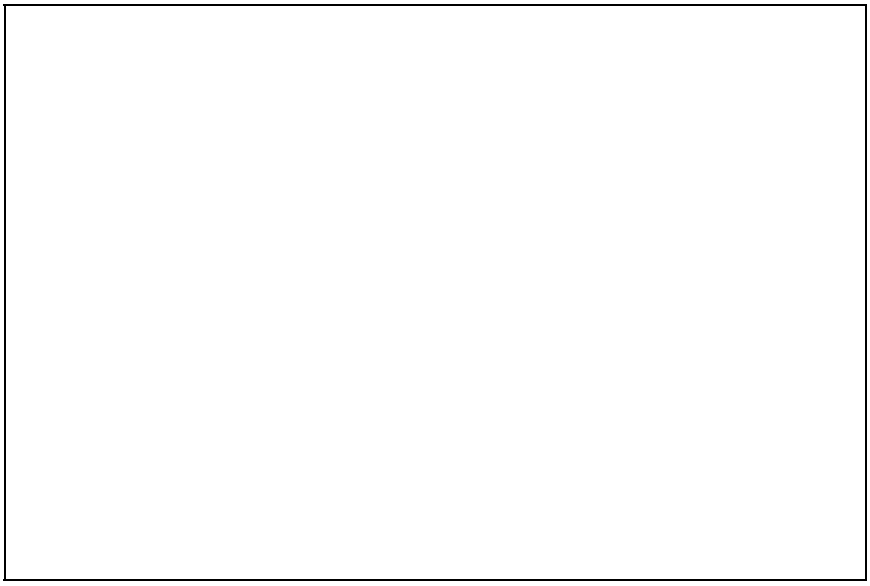
The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

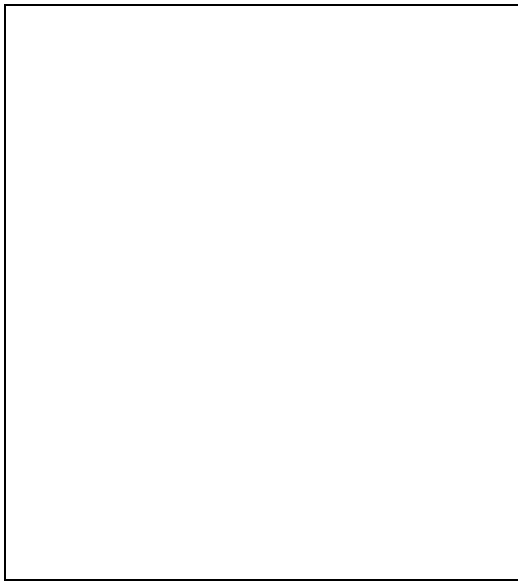
The plot on the right shows the iteration times for the two measurements. Again, the red dots represent the previous measurement while the blue dots show the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Reads/minio: one-node/1KiB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Slope	1.2538 ms	1.2701 ms	1.2869 ms
Throughput	777.08 KiB/s	787.35 KiB/s	797.57 KiB/s
R²	0.8254551	0.8354323	0.8248011
Mean	1.2538 ms	1.2655 ms	1.2779 ms
Std. Dev.	49.524 Âµs	61.860 Âµs	73.273 Âµs
Median	1.2367 ms	1.2513 ms	1.2652 ms
MAD	34.711 Âµs	50.198 Âµs	66.424 Âµs

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD
- Slope

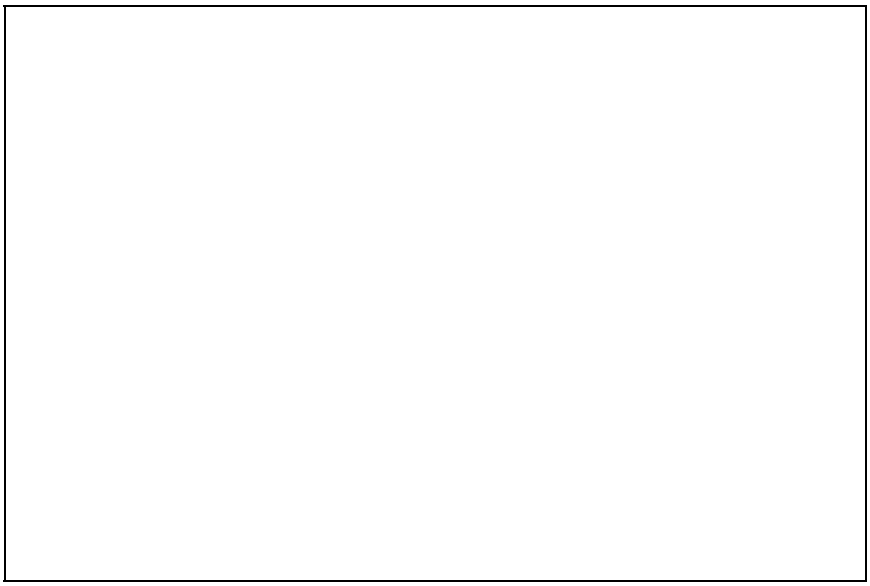
Understanding this report:

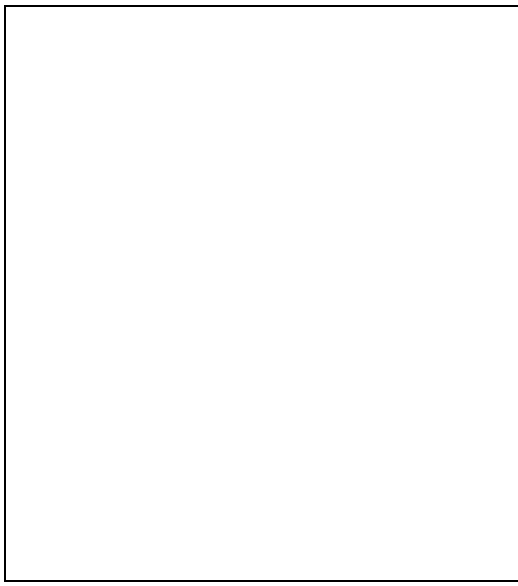
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the linear regression calculated from the measurements. Each point represents a sample, though here it shows the total time for the sample rather than time per iteration. The line is the line of best fit for these measurements.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	-2.9670%	-1.7873%	-0.4344%	(p = 0.01 < 0.05)
Change in throughput	+3.0577%	+1.8199%	+0.4363%	

Change within noise threshold.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

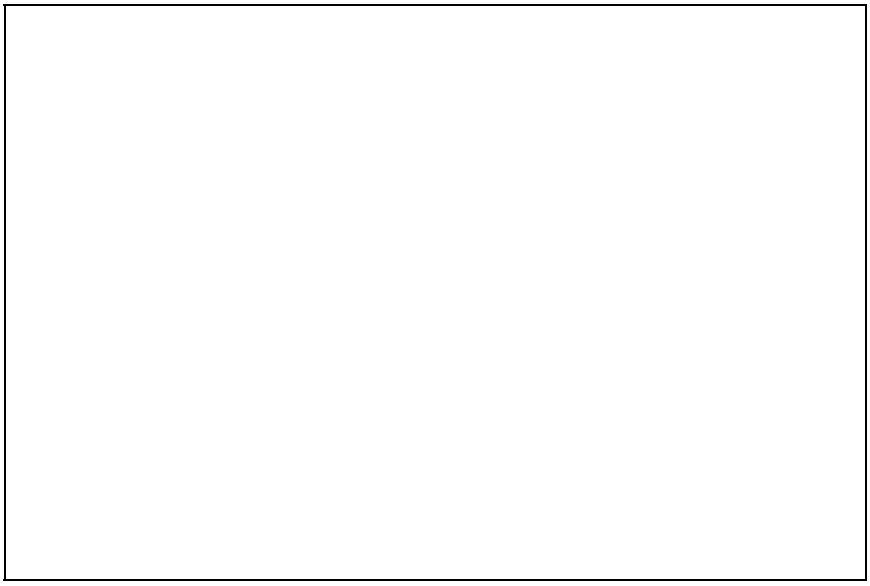
The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

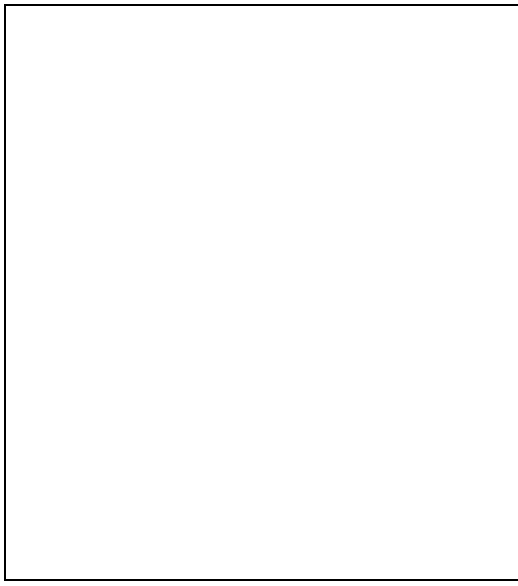
The plot on the right shows the two regressions. Again, the red line represents the previous measurement while the blue line shows the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Reads/minio: one-node/1MiB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Throughput	111.62 MiB/s	113.65 MiB/s	115.72 MiB/s
R²	0.0029233	0.0030339	0.0029203
Mean	8.6414 ms	8.7991 ms	8.9590 ms
Std. Dev.	701.20 Âµs	813.76 Âµs	921.13 Âµs
Median	8.5506 ms	8.6932 ms	9.0168 ms
MAD	679.65 Âµs	909.45 Âµs	1.1517 ms

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD

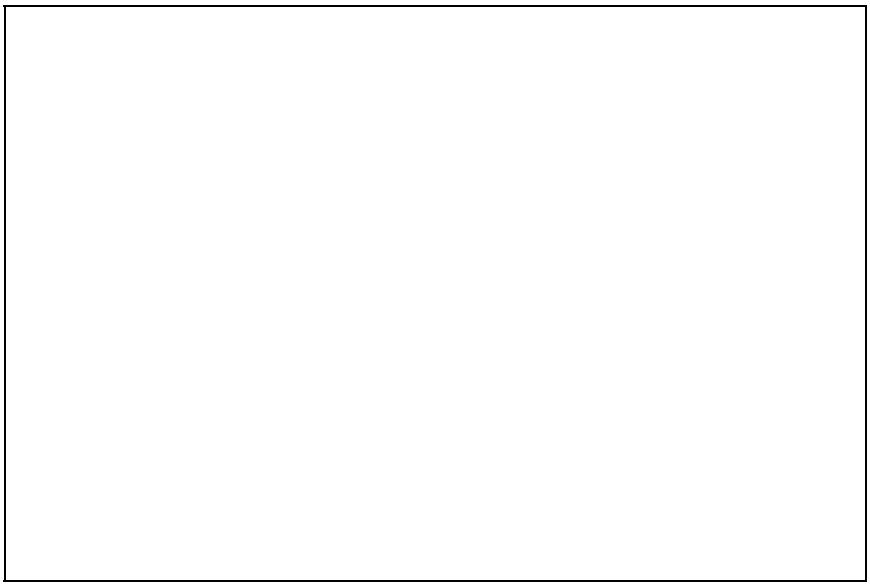
Understanding this report:

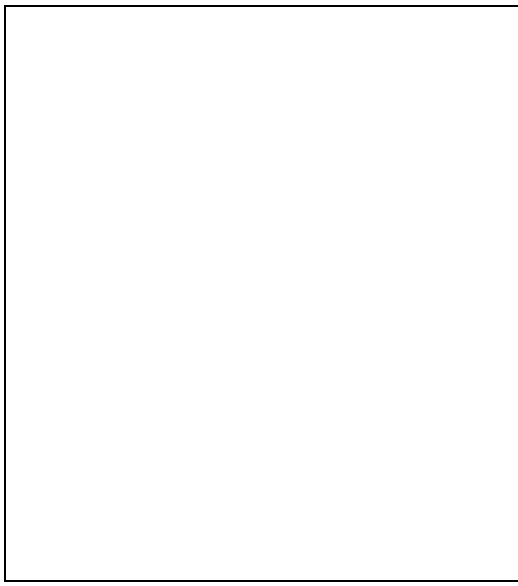
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the average time per iteration for the samples. Each point represents one sample.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	+4.9666%	+7.6326%	+10.515%	(p = 0.00 < 0.05)
Change in throughput	-4.7316%	-7.0914%	-9.5144%	

Performance has regressed.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

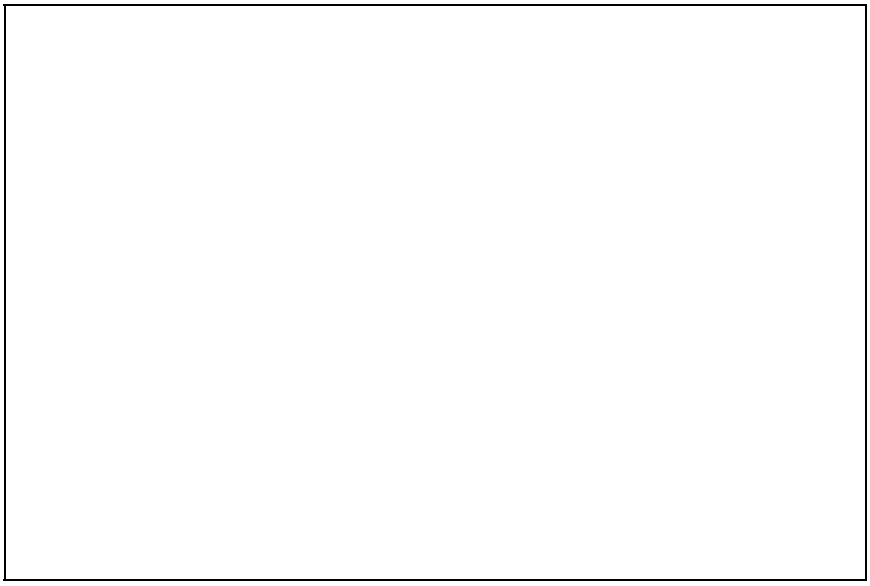
The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

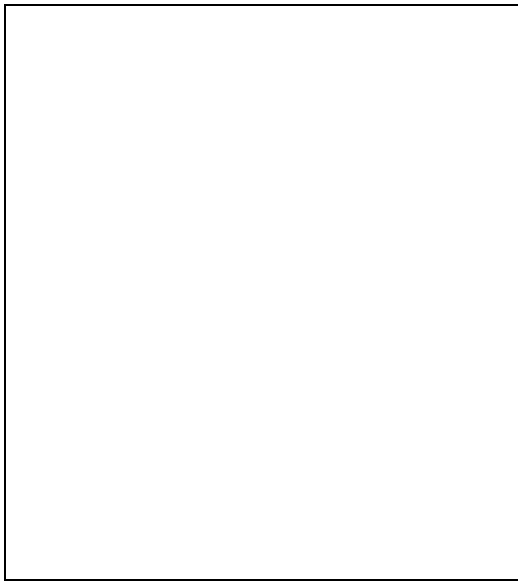
The plot on the right shows the iteration times for the two measurements. Again, the red dots represent the previous measurement while the blue dots show the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Reads/minio: one-node/50KiB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Slope	1.9505 ms	2.3261 ms	2.7199 ms
Throughput	17.952 MiB/s	20.991 MiB/s	25.034 MiB/s
R²	0.0163594	0.0176616	0.0162407
Mean	1.7417 ms	1.9142 ms	2.1107 ms
Std. Dev.	642.03 Âµs	947.67 Âµs	1.1787 ms
Median	1.5841 ms	1.5981 ms	1.6081 ms
MAD	41.519 Âµs	55.851 Âµs	92.209 Âµs

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD
- Slope

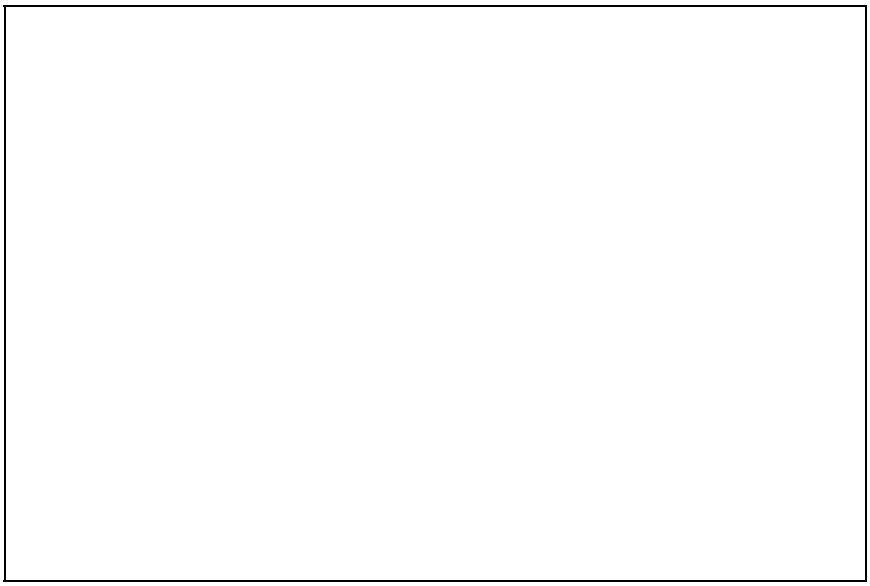
Understanding this report:

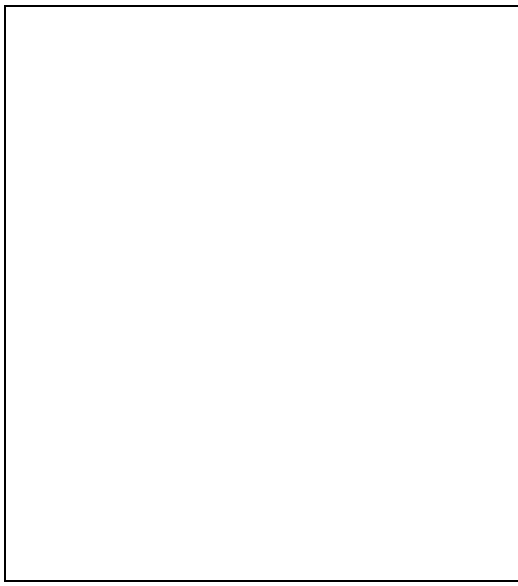
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the linear regression calculated from the measurements. Each point represents a sample, though here it shows the total time for the sample rather than time per iteration. The line is the line of best fit for these measurements.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	-11.644%	+1.3882%	+16.095%	(p = 0.85 > 0.05)
Change in throughput	+13.179%	-1.3692%	-13.863%	

No change in performance detected.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

The plot on the right shows the two regressions. Again, the red line represents the previous measurement while the blue line shows the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Reads/minio: one-node

Violin Plot

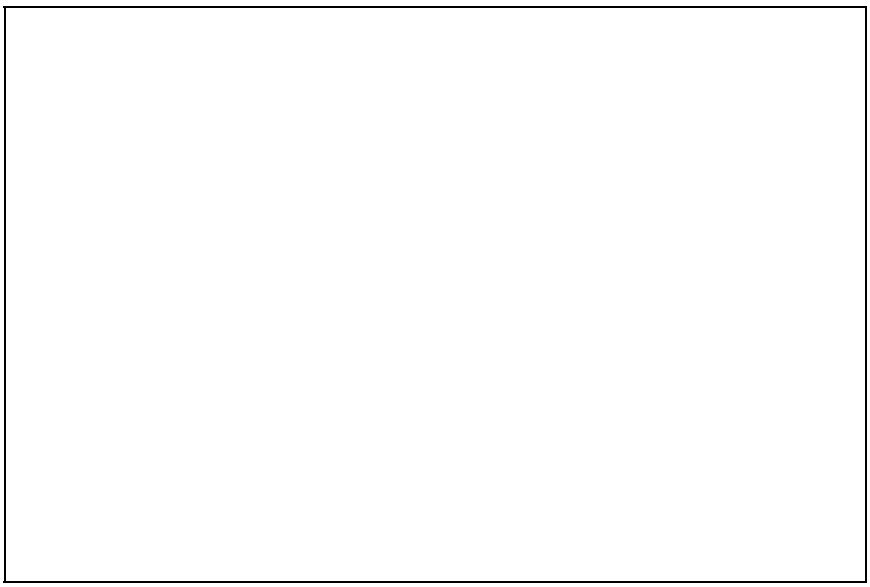
This chart shows the relationship between function/parameter and iteration time. The thickness of the shaded region indicates the probability that a measurement of the given function/parameter would take a particular length of time.

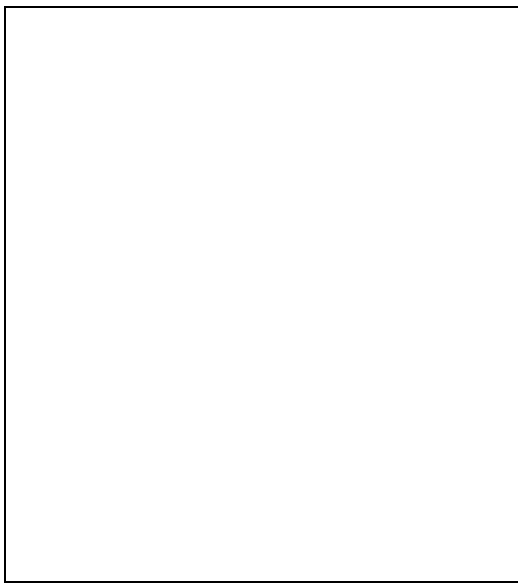
Line Chart

This chart shows the mean measured time for each function as the input (or the size of the input) increases.

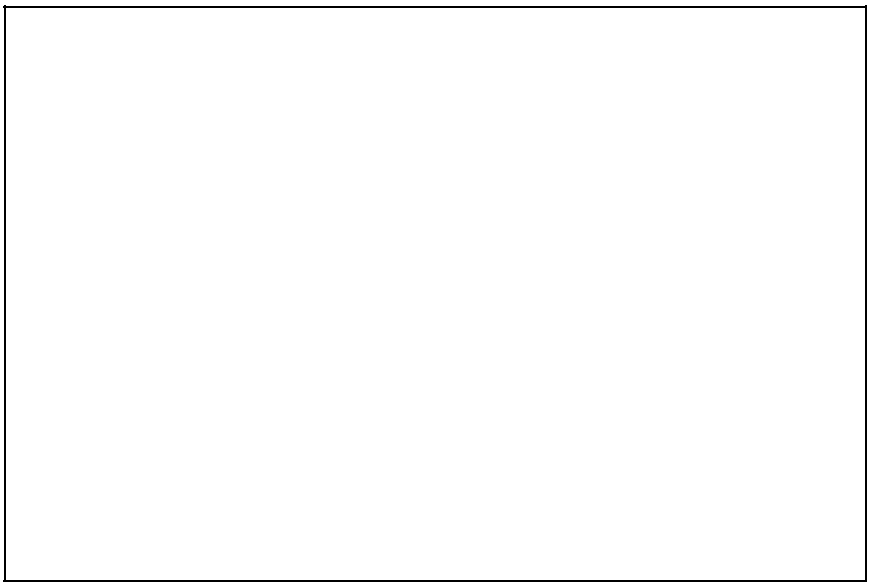
Reads/minio: one-node/1KiB

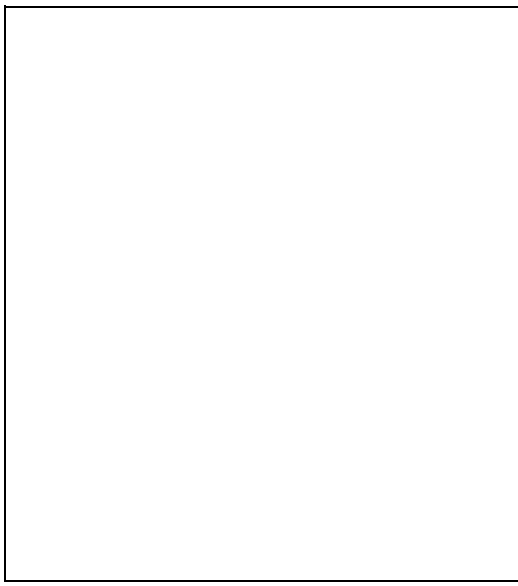
Reads/minio: one-node/50KiB



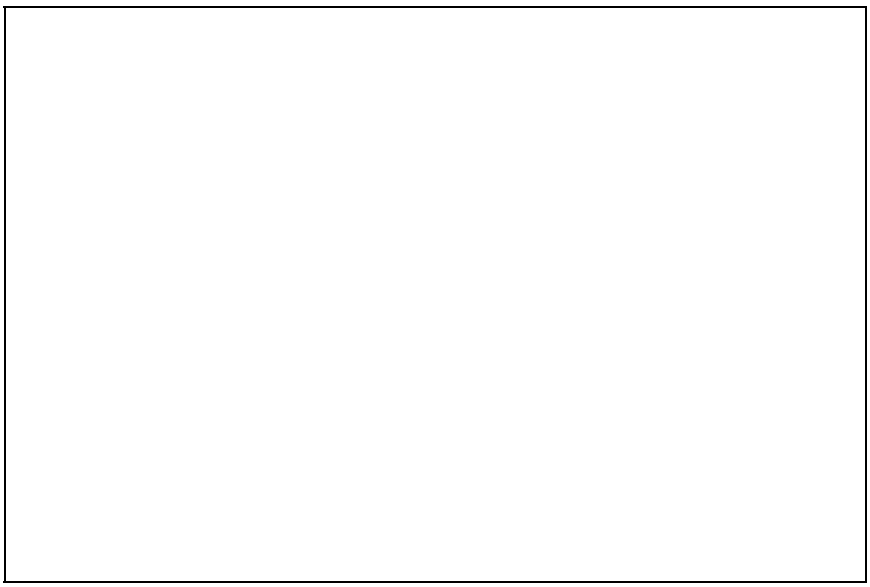


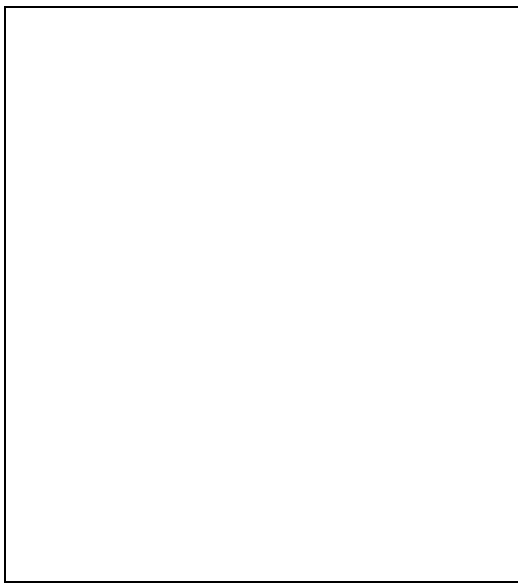
Reads/minio: one-node/1MiB





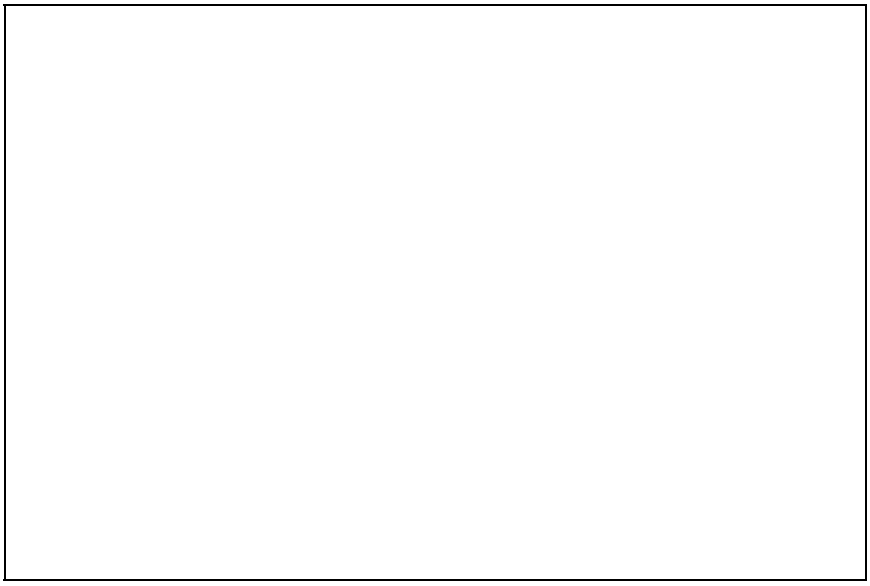
Reads/minio: one-node/100MiB

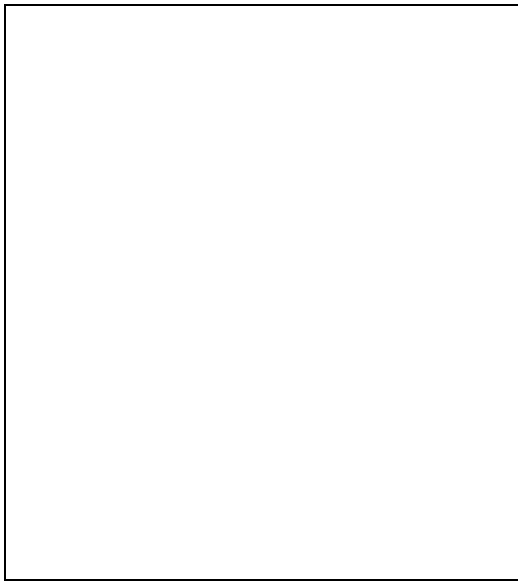




This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Reads/minio: two-node/0B





Additional Statistics:

	Lower bound	Estimate	Upper bound
Throughput	488.40 KiB/s	495.73 KiB/s	502.10 KiB/s
R²	0.0023368	0.0024111	0.0023084
Mean	1.9916 ms	2.0172 ms	2.0475 ms
Std. Dev.	85.950 Âµs	144.15 Âµs	196.41 Âµs
Median	1.9628 ms	1.9871 ms	1.9975 ms
MAD	46.315 Âµs	58.834 Âµs	74.367 Âµs

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD

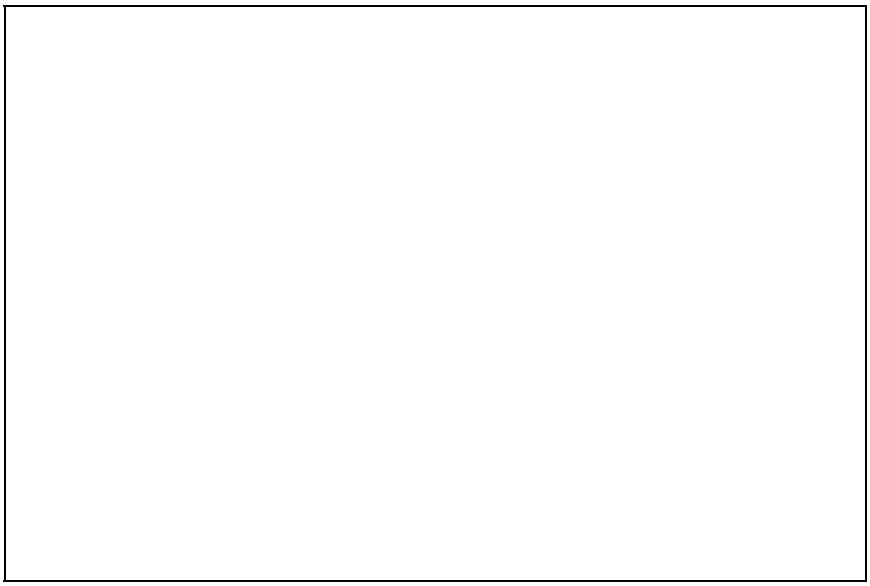
Understanding this report:

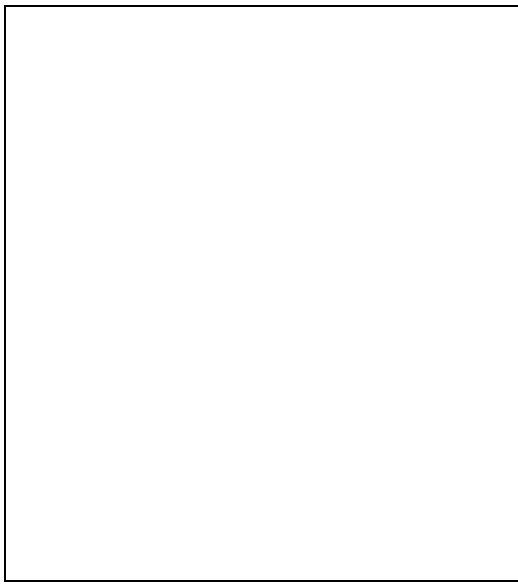
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the average time per iteration for the samples. Each point represents one sample.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	-65.612%	-64.106%	-62.573%	(p = 0.00 < 0.05)
Change in throughput	+190.80%	+178.59%	+167.18%	

Performance has improved.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

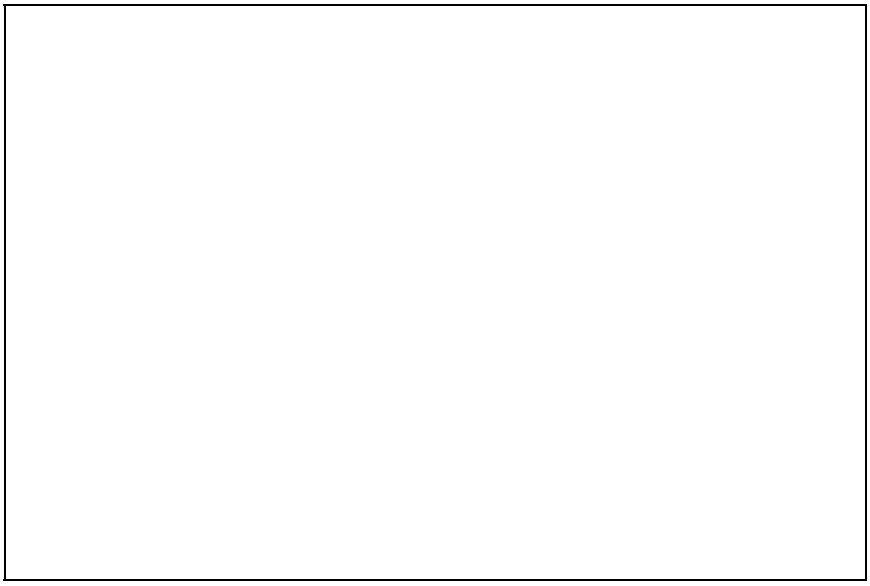
The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

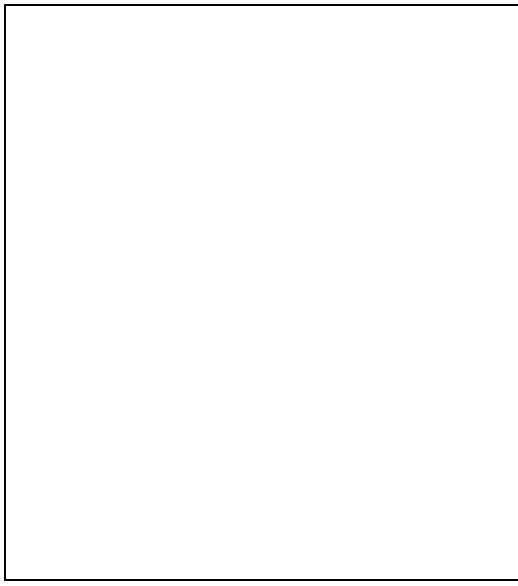
The plot on the right shows the iteration times for the two measurements. Again, the red dots represent the previous measurement while the blue dots show the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Reads/minio: two-node/0B #2





Additional Statistics:

	Lower bound	Estimate	Upper bound
Throughput	18.285 MiB/s	18.711 MiB/s	19.028 MiB/s
R²	0.0000607	0.0000622	0.0000593
Mean	2.5661 ms	2.6096 ms	2.6704 ms
Std. Dev.	71.142 Âµs	274.94 Âµs	436.85 Âµs
Median	2.5385 ms	2.5635 ms	2.5812 ms
MAD	50.566 Âµs	67.867 Âµs	81.068 Âµs

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD

Understanding this report:

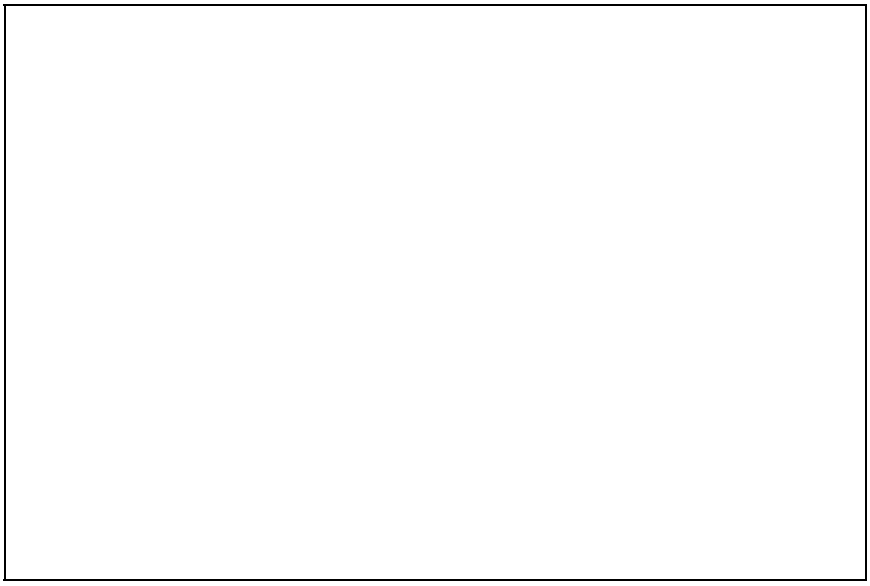
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

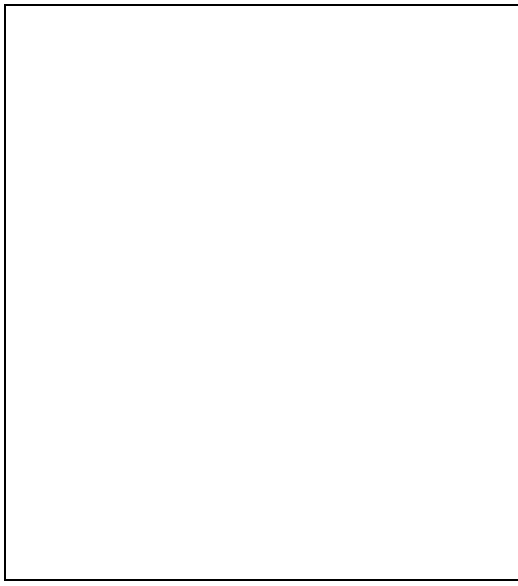
The plot on the right shows the average time per iteration for the samples. Each point represents one sample.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Reads/minio: two-node/0B #3





Additional Statistics:

	Lower bound	Estimate	Upper bound
Throughput	253.03 MiB/s	259.08 MiB/s	264.76 MiB/s
R²	0.0087507	0.0090452	0.0086839
Mean	3.7770 ms	3.8599 ms	3.9521 ms
Std. Dev.	322.14 Âµs	452.17 Âµs	569.19 Âµs
Median	3.6547 ms	3.7104 ms	3.7704 ms
MAD	149.82 Âµs	215.59 Âµs	272.99 Âµs

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD

Understanding this report:

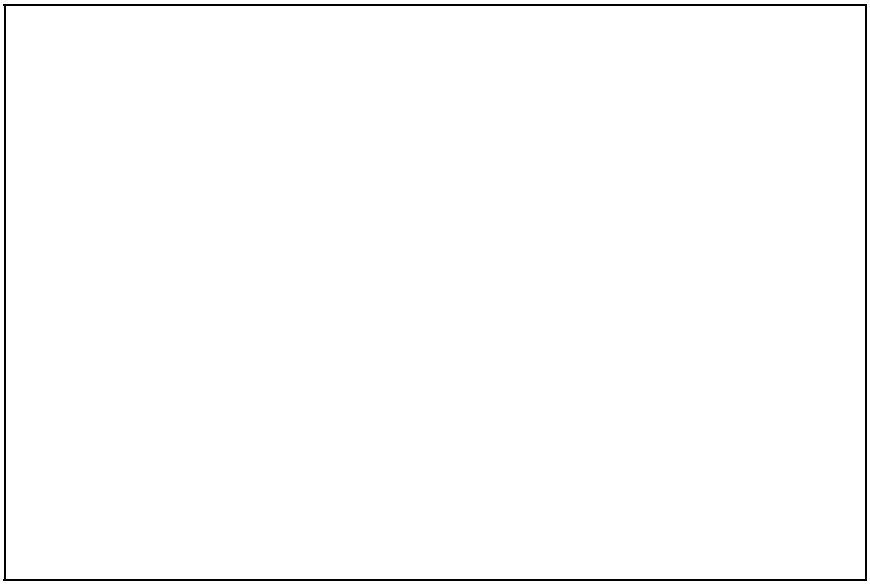
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

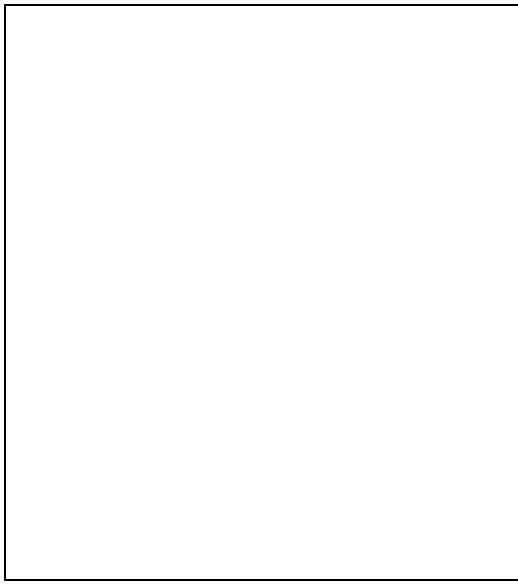
The plot on the right shows the average time per iteration for the samples. Each point represents one sample.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Reads/minio: two-node/0B #4





Additional Statistics:

	Lower bound	Estimate	Upper bound
Throughput	17.099 GiB/s	17.835 GiB/s	18.555 GiB/s
R²	0.0010584	0.0010948	0.0010503
Mean	5.2631 ms	5.4754 ms	5.7111 ms
Std. Dev.	819.12 Âµs	1.1501 ms	1.4528 ms
Median	4.9610 ms	5.0916 ms	5.3149 ms
MAD	492.16 Âµs	615.74 Âµs	860.89 Âµs

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD

Understanding this report:

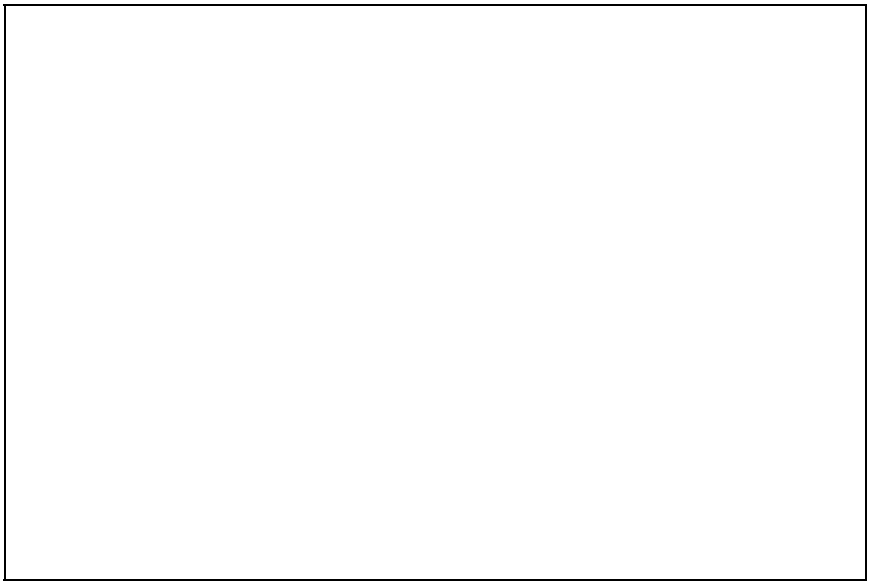
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

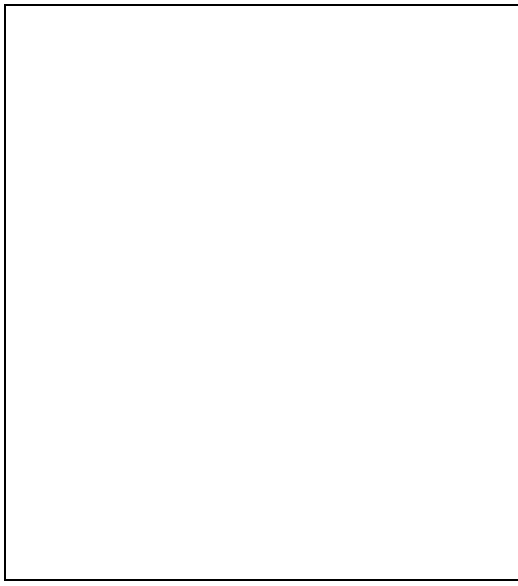
The plot on the right shows the average time per iteration for the samples. Each point represents one sample.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Reads/minio: two-node/100MiB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Throughput	204.97 MiB/s	207.98 MiB/s	210.67 MiB/s
R^2	0.0002211	0.0002285	0.0002189
Mean	474.67 ms	480.82 ms	487.87 ms
Std. Dev.	21.385 ms	33.786 ms	43.480 ms
Median	471.14 ms	475.01 ms	482.75 ms
MAD	14.545 ms	19.753 ms	22.493 ms

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD

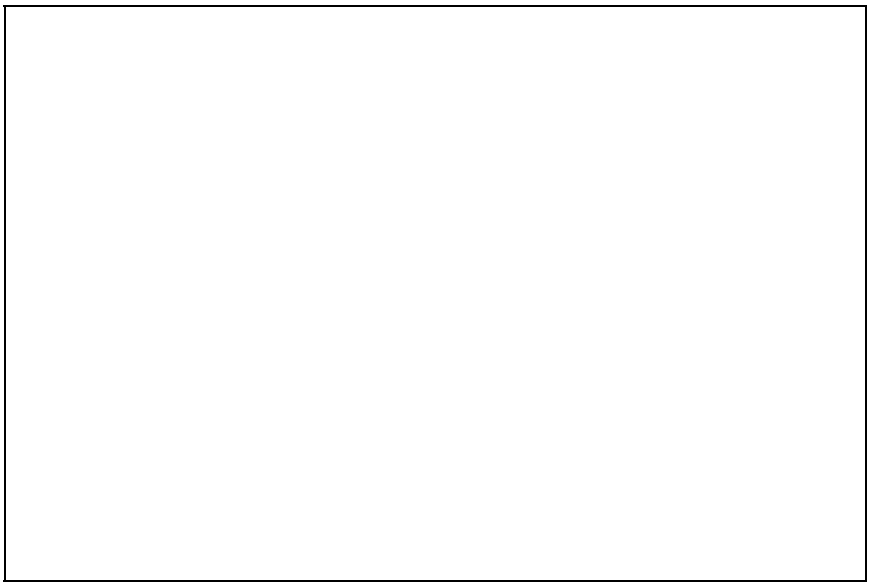
Understanding this report:

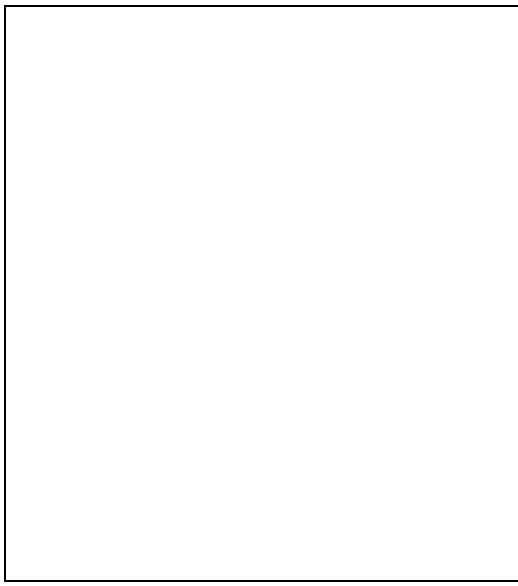
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the average time per iteration for the samples. Each point represents one sample.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	+8237.5%	+8582.3%	+8929.7%	(p = 0.00 < 0.05)
Change in throughput	-98.801%	-98.848%	-98.893%	

Performance has regressed.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

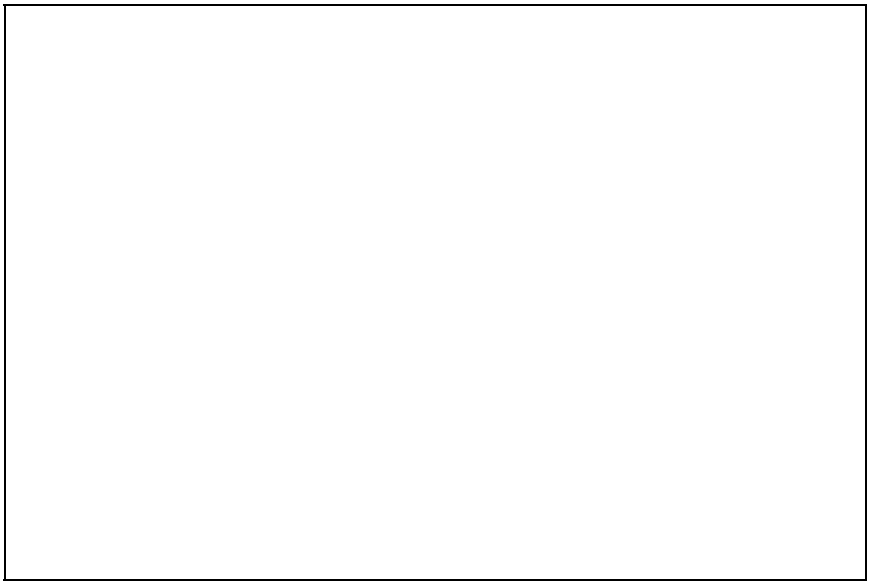
The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

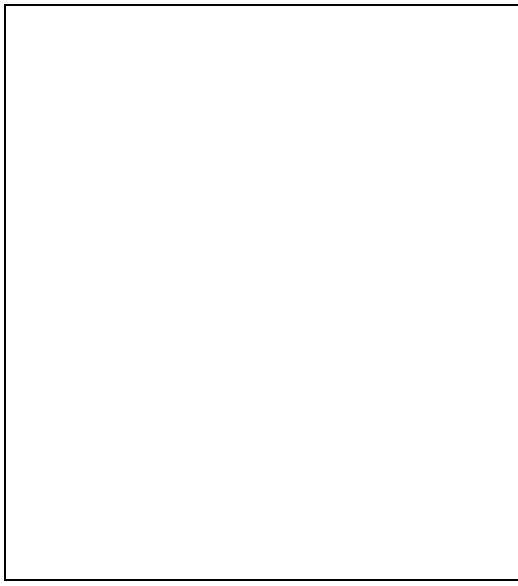
The plot on the right shows the iteration times for the two measurements. Again, the red dots represent the previous measurement while the blue dots show the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Reads/minio: two-node/1KiB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Throughput	492.75 KiB/s	498.94 KiB/s	504.02 KiB/s
R²	0.0006550	0.0006746	0.0006445
Mean	1.9841 ms	2.0042 ms	2.0294 ms
Std. Dev.	62.481 Âµs	117.30 Âµs	174.06 Âµs
Median	1.9798 ms	1.9875 ms	1.9934 ms
MAD	25.946 Âµs	38.784 Âµs	50.220 Âµs

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD

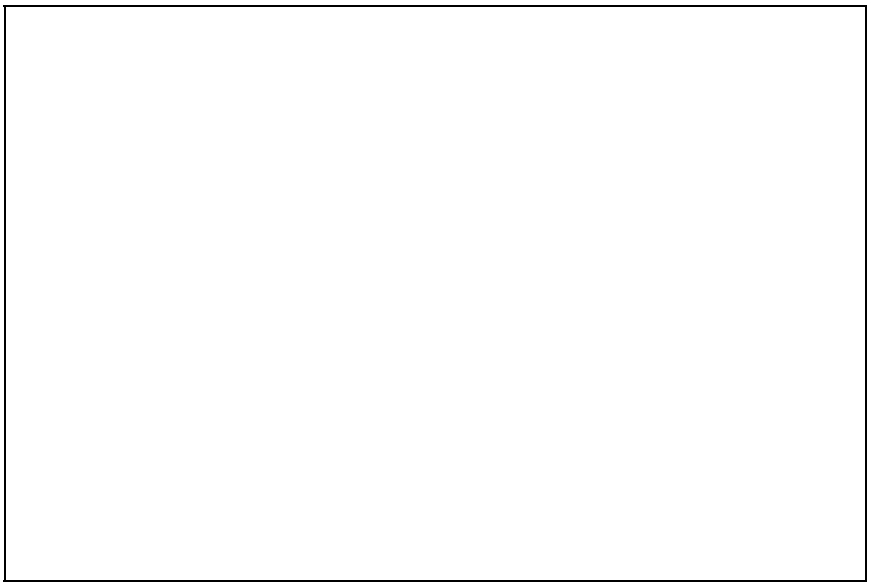
Understanding this report:

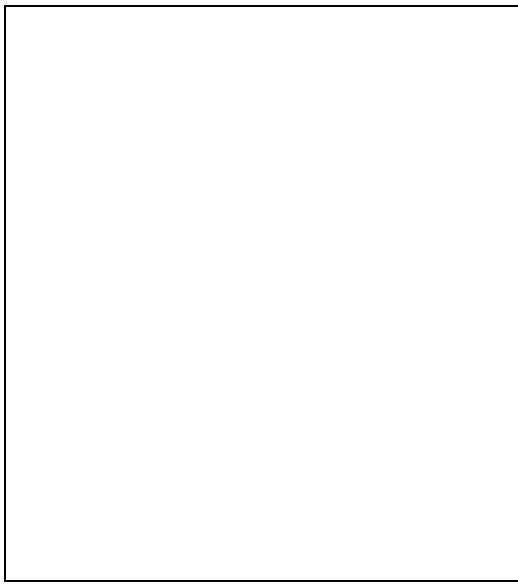
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the average time per iteration for the samples. Each point represents one sample.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	-0.7729%	+0.7559%	+2.2833%	(p = 0.37 > 0.05)
Change in throughput	+0.7789%	-0.7502%	-2.2323%	

No change in performance detected.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

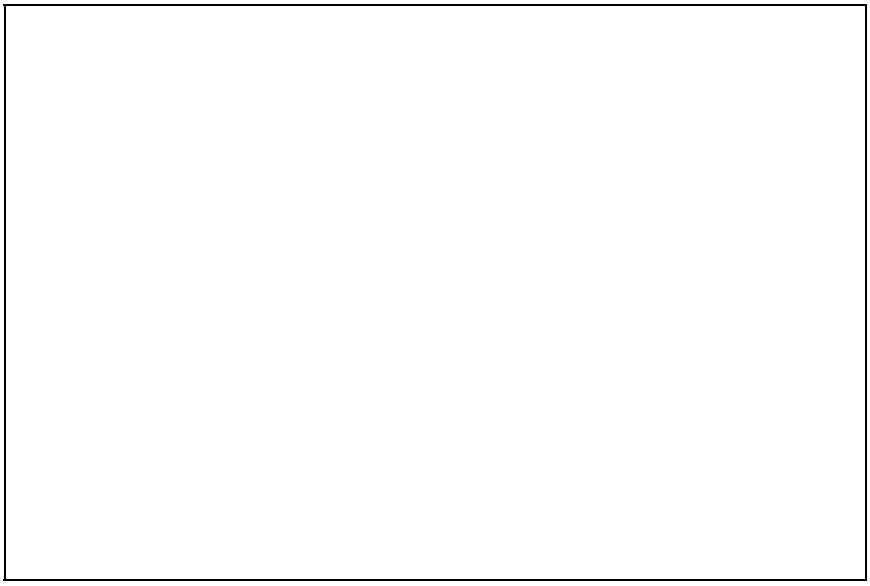
The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

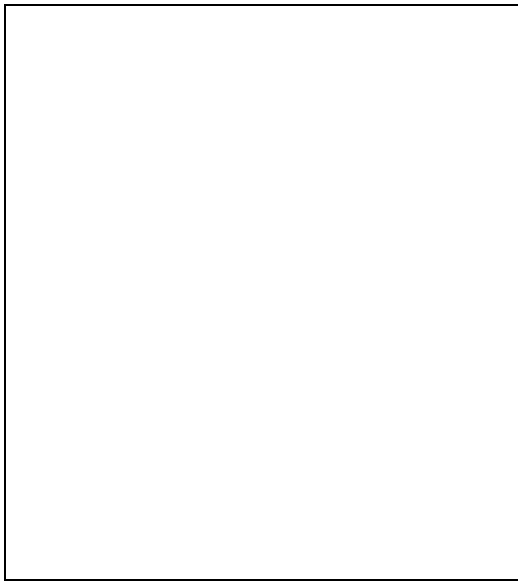
The plot on the right shows the iteration times for the two measurements. Again, the red dots represent the previous measurement while the blue dots show the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Reads/minio: two-node/1MiB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Throughput	107.88 MiB/s	109.83 MiB/s	111.91 MiB/s
R²	0.0024746	0.0025722	0.0024792
Mean	8.9354 ms	9.1046 ms	9.2695 ms
Std. Dev.	769.19 Âµs	854.67 Âµs	926.27 Âµs
Median	8.9602 ms	9.4001 ms	9.5677 ms
MAD	652.12 Âµs	918.15 Âµs	1.2303 ms

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD

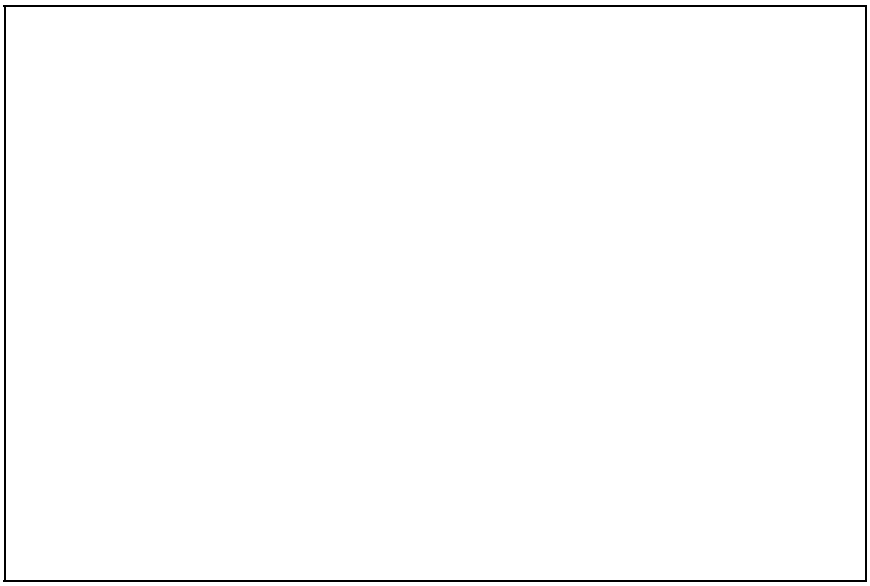
Understanding this report:

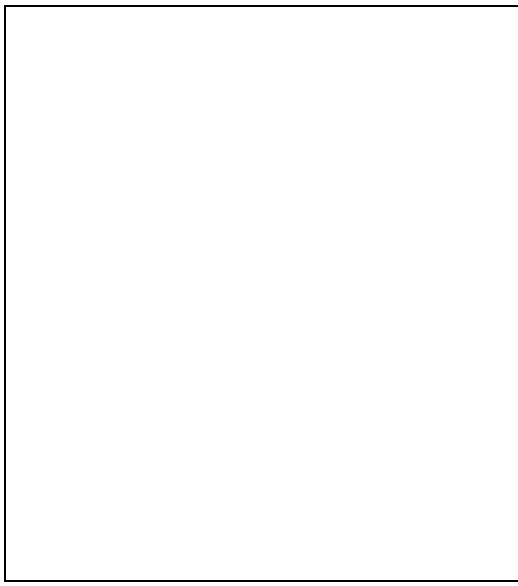
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the average time per iteration for the samples. Each point represents one sample.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	+2.6726%	+5.4663%	+8.2405%	(p = 0.00 < 0.05)
Change in throughput	-2.6030%	-5.1830%	-7.6131%	

Performance has regressed.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

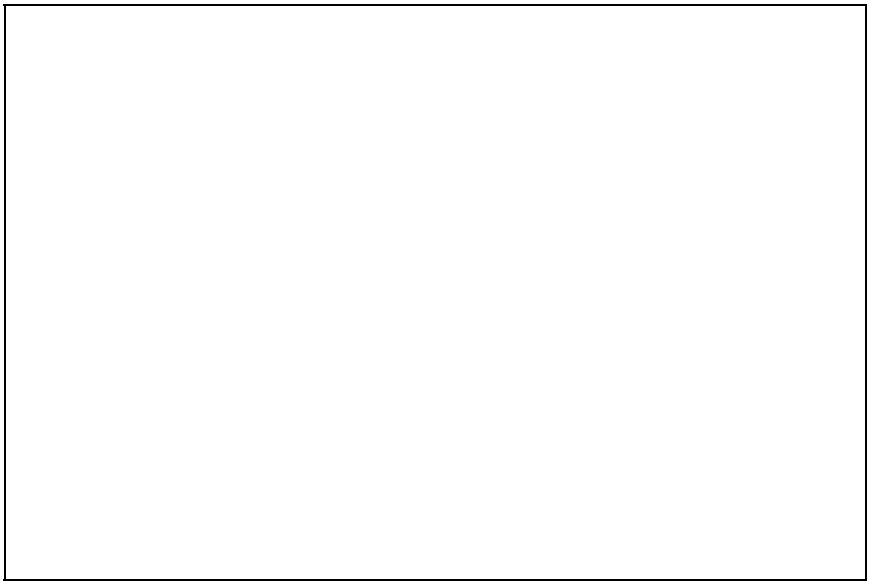
The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

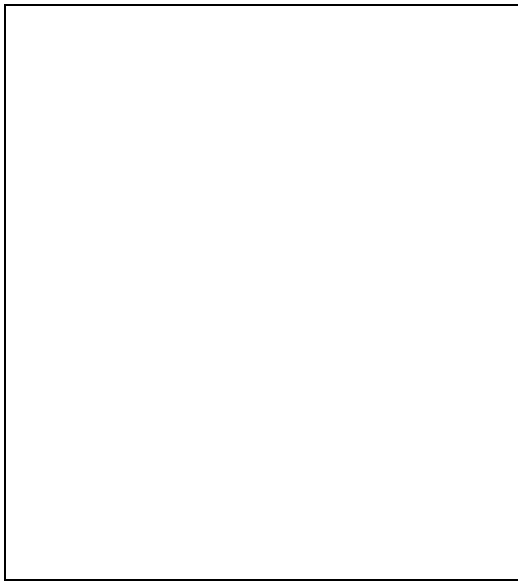
The plot on the right shows the iteration times for the two measurements. Again, the red dots represent the previous measurement while the blue dots show the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Reads/minio: two-node/50KiB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Throughput	20.029 MiB/s	20.243 MiB/s	20.442 MiB/s
R²	0.0027567	0.0028518	0.0027368
Mean	2.3886 ms	2.4120 ms	2.4379 ms
Std. Dev.	94.229 Âµs	126.74 Âµs	154.37 Âµs
Median	2.3543 ms	2.3769 ms	2.3991 ms
MAD	55.284 Âµs	75.345 Âµs	95.142 Âµs

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD

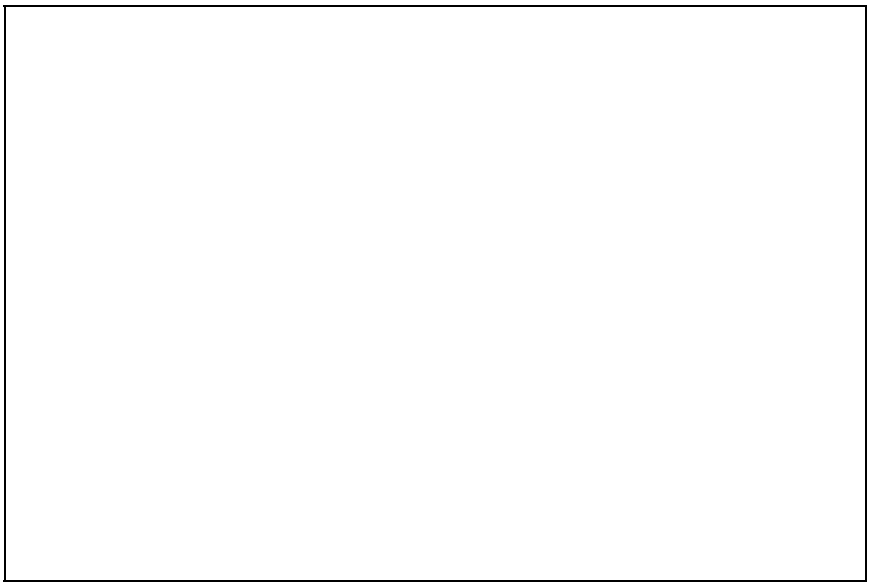
Understanding this report:

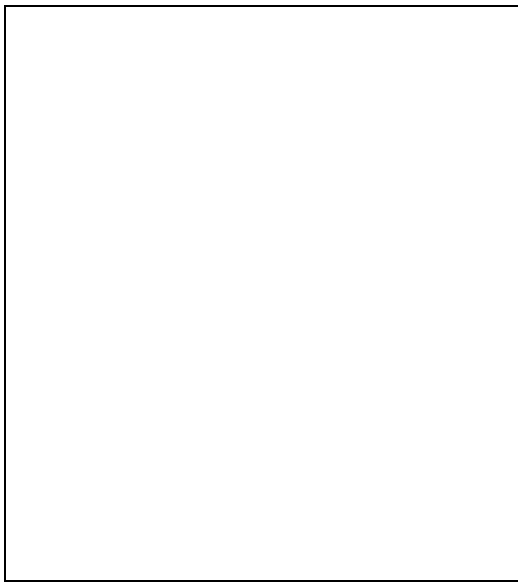
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the average time per iteration for the samples. Each point represents one sample.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	-2.0950%	-0.3151%	+1.4750%	(p = 0.73 > 0.05)
Change in throughput	+2.1399%	+0.3161%	-1.4536%	

No change in performance detected.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

The plot on the right shows the iteration times for the two measurements. Again, the red dots represent the previous measurement while the blue dots show the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Reads/minio: two-node

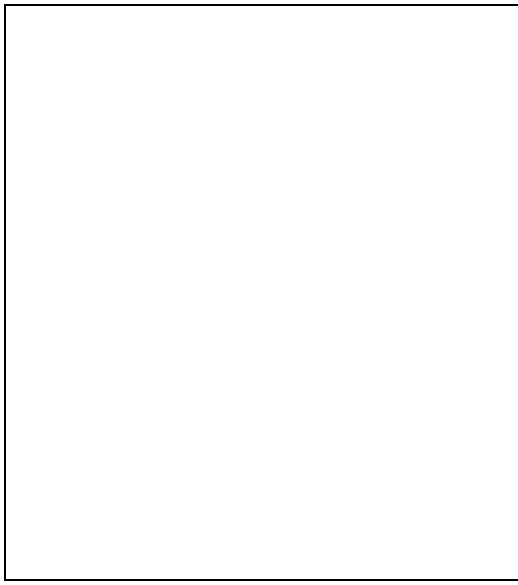
Violin Plot

This chart shows the relationship between function/parameter and iteration time. The thickness of the shaded region indicates the probability that a measurement of the given function/parameter would take a particular length of time.

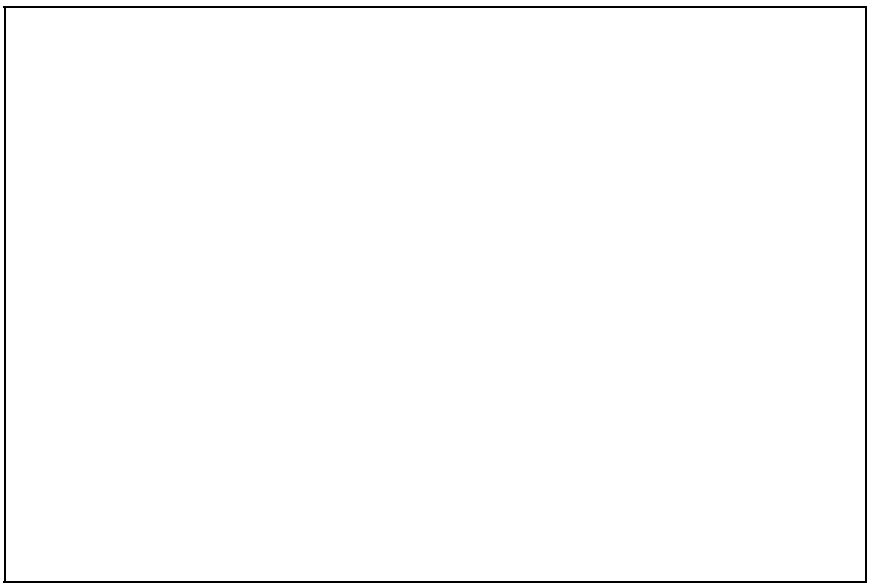
Line Chart

This chart shows the mean measured time for each function as the input (or the size of the input) increases.

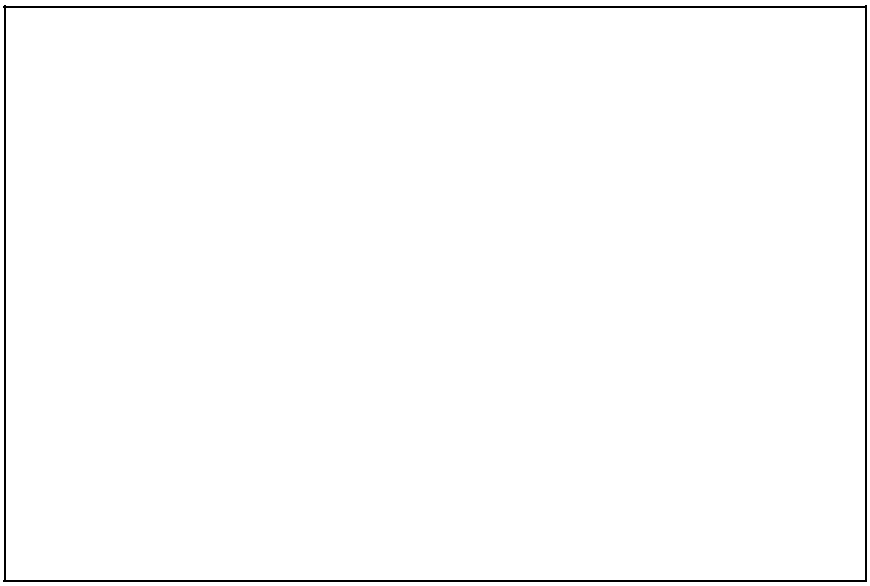
Reads/minio: two-node/1KiB



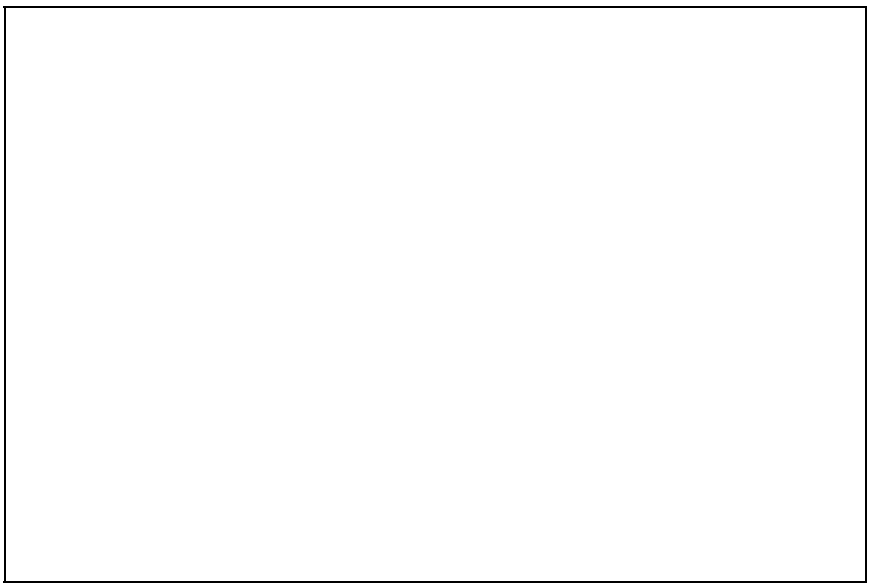
Reads/minio: two-node/50KiB

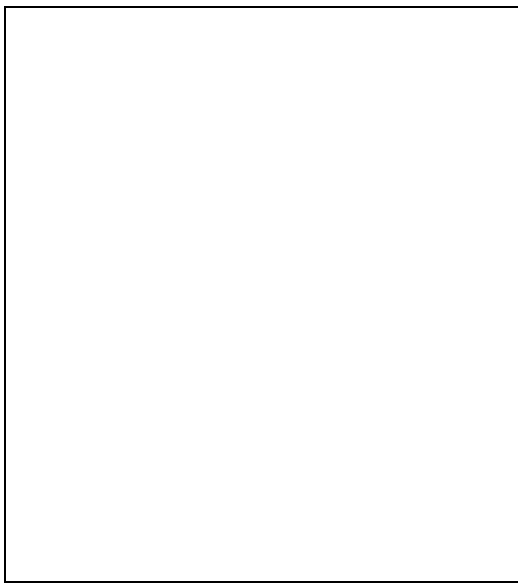


Reads/minio: two-node/1MiB



Reads/minio: two-node/100MiB





This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Reads

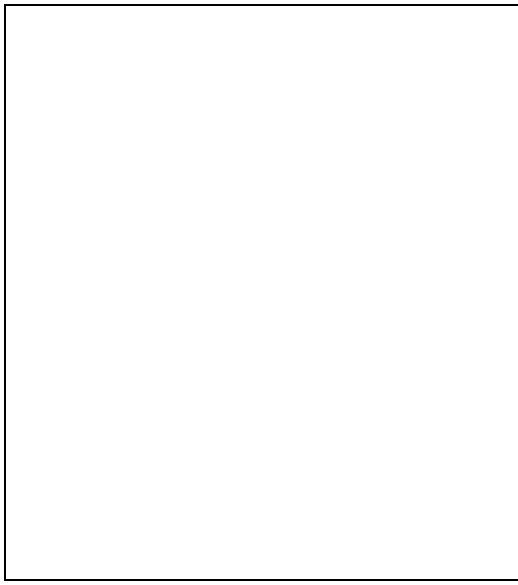
Violin Plot

This chart shows the relationship between function/parameter and iteration time. The thickness of the shaded region indicates the probability that a measurement of the given function/parameter would take a particular length of time.

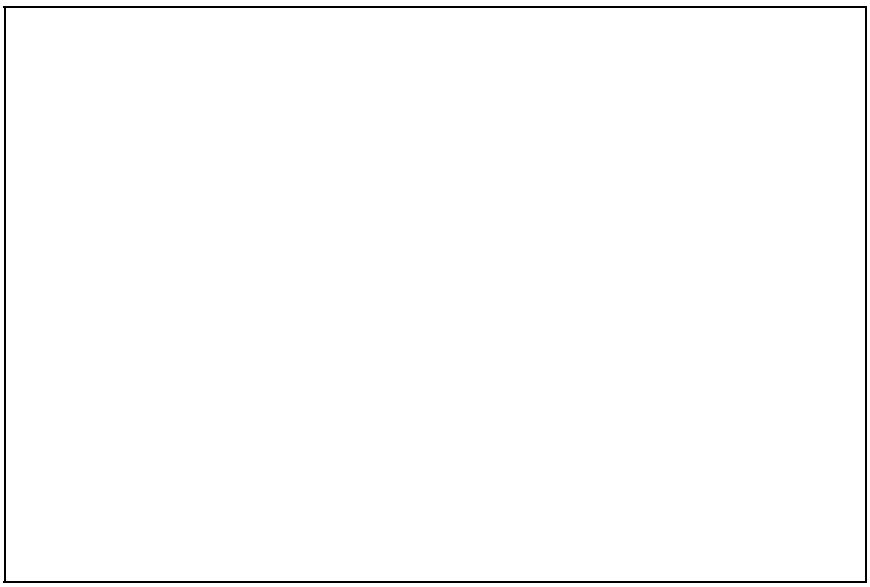
Line Chart

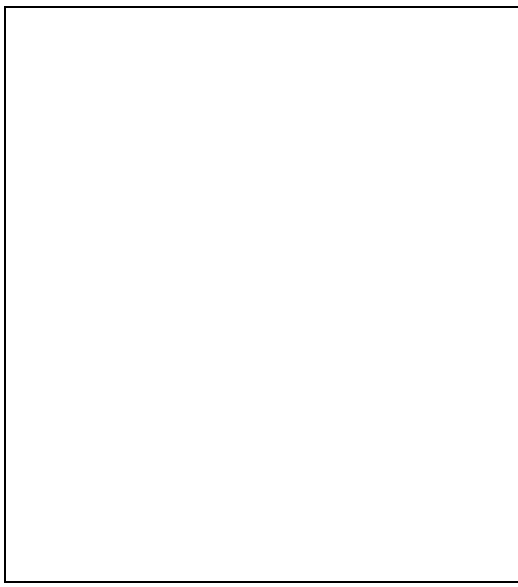
This chart shows the mean measured time for each function as the input (or the size of the input) increases.

Reads/bob: one-node/100MiB

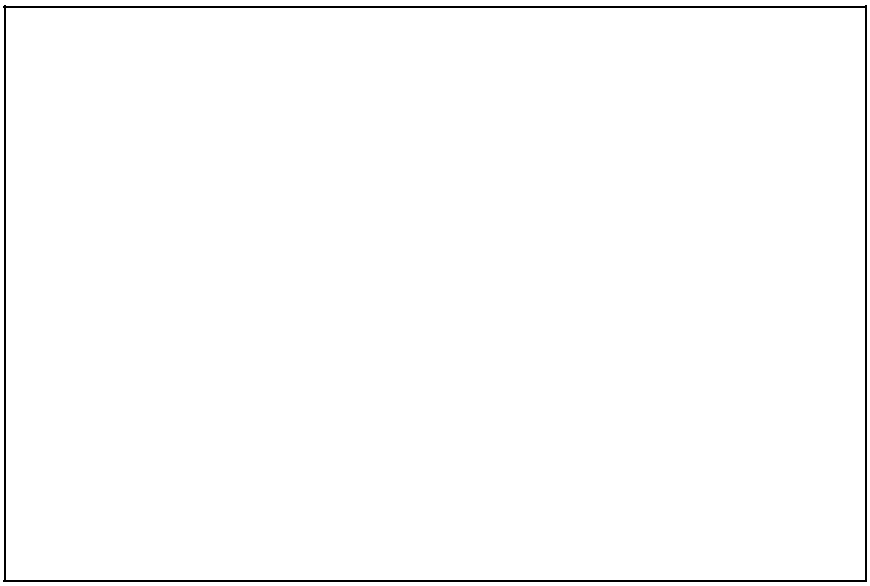


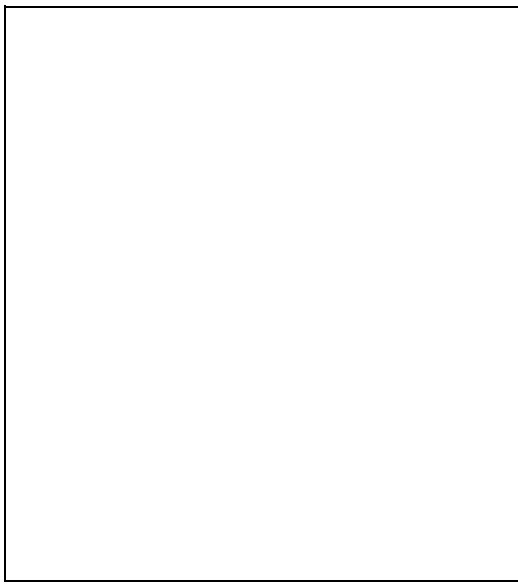
Reads/bob: one-node/1KiB



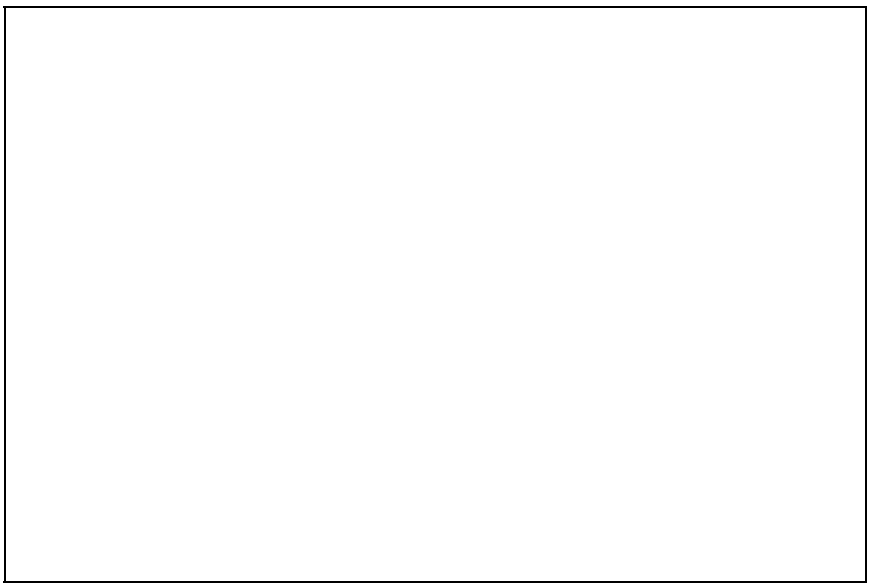


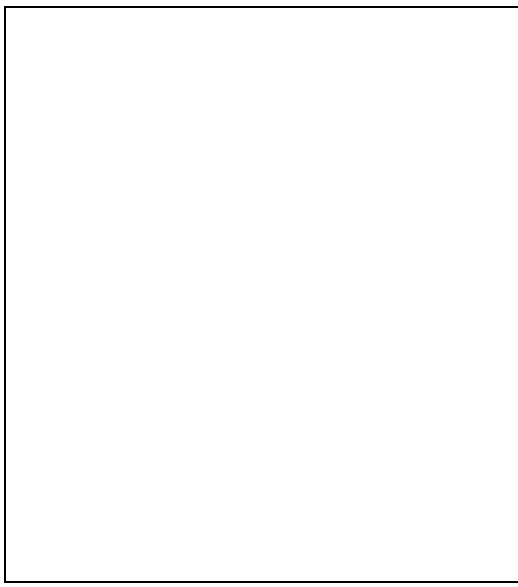
Reads/bob: one-node/1MiB



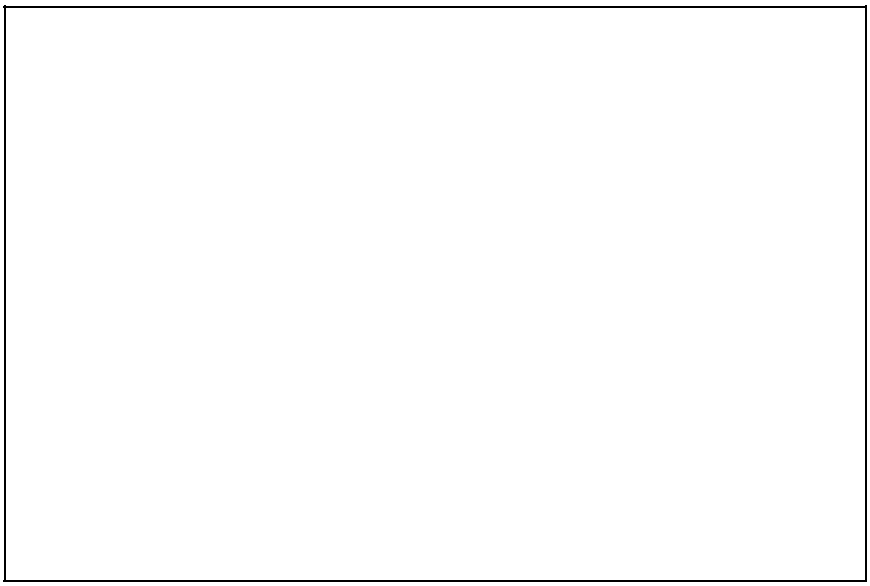


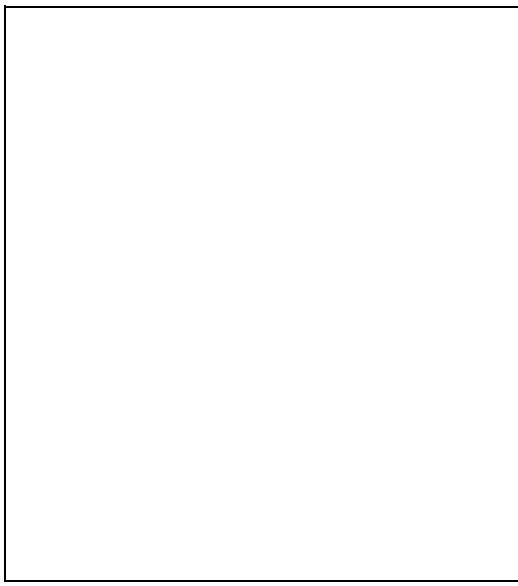
Reads/bob: one-node/50KiB



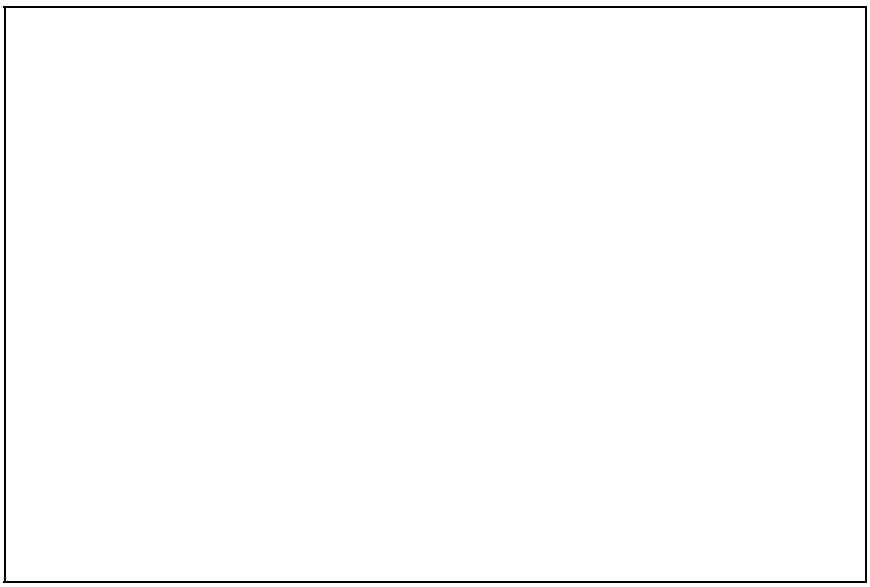


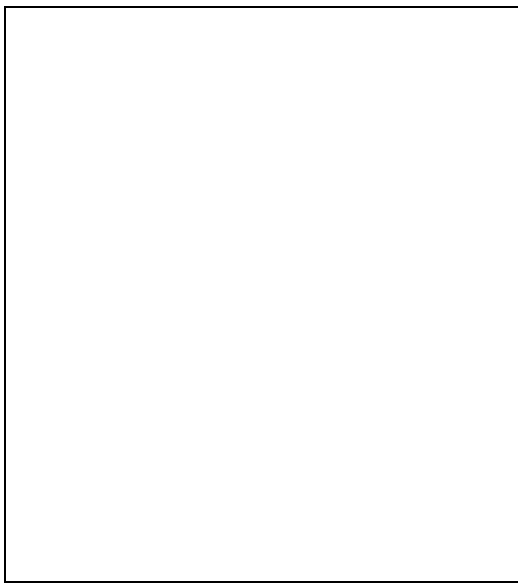
Reads/bob: two-node/100MiB



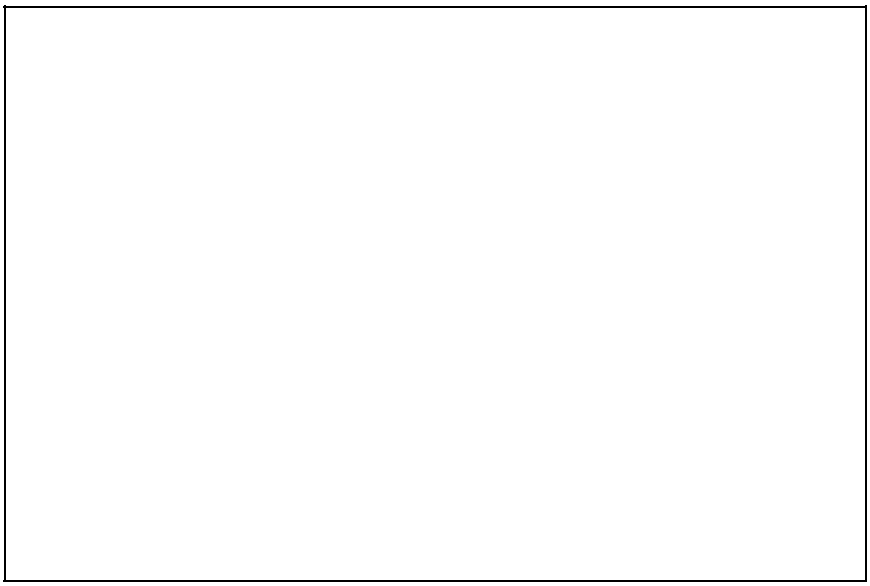


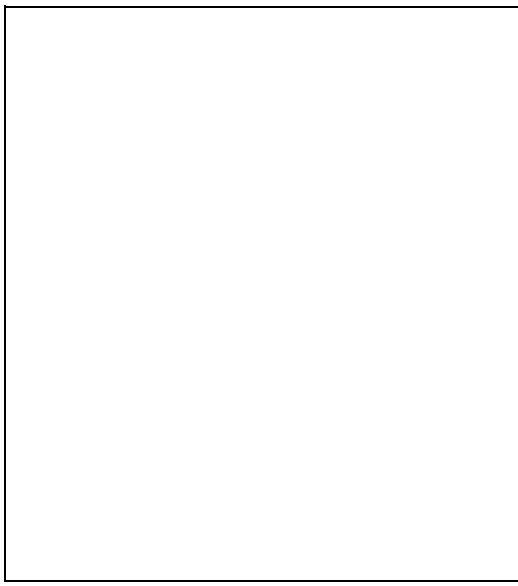
Reads/bob: two-node/1KiB



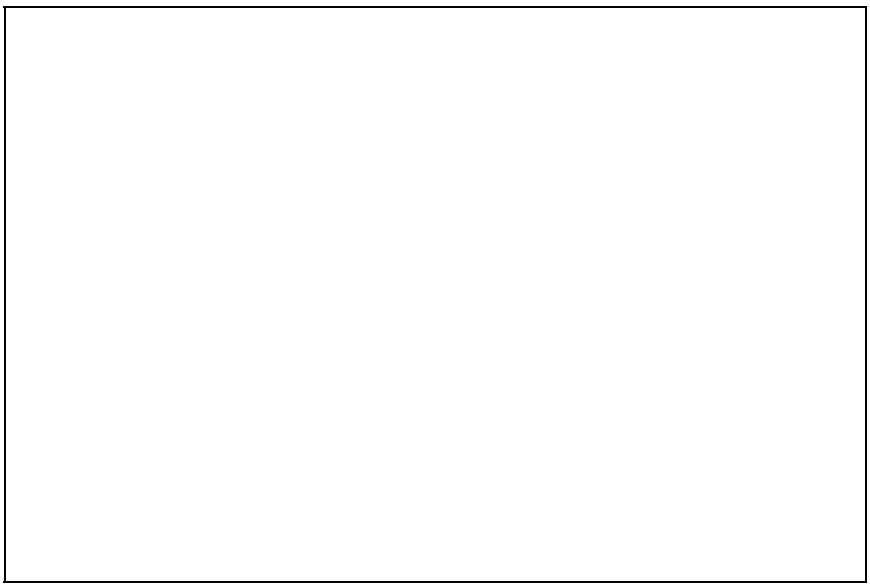


Reads/bob: two-node/1MiB

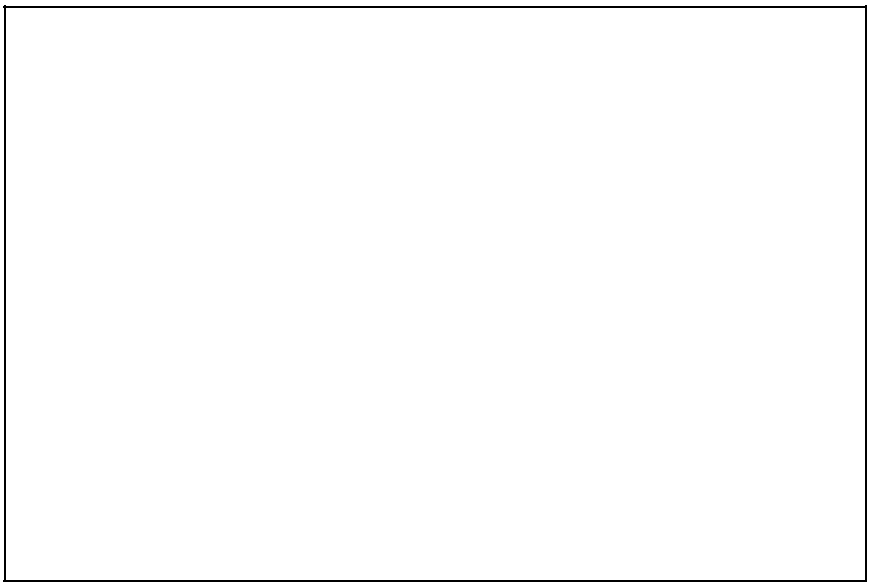




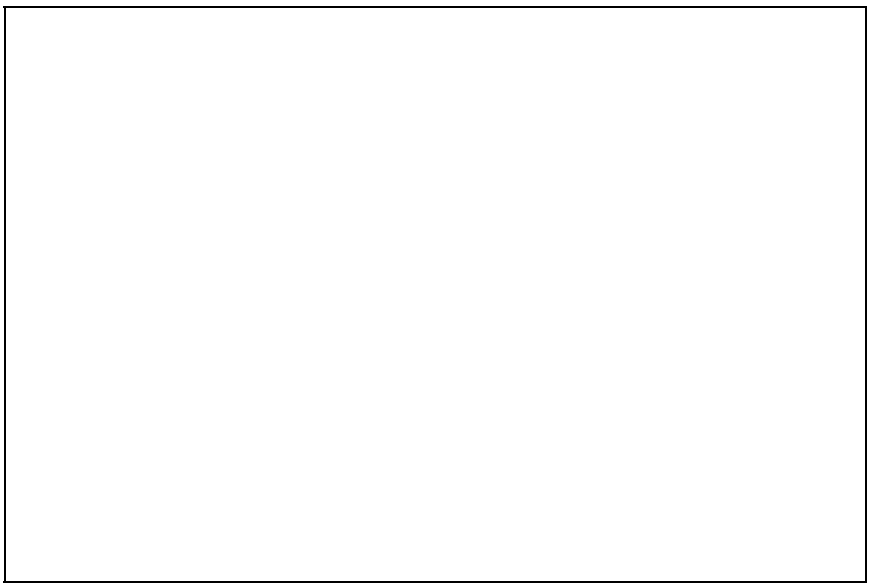
Reads/bob: two-node/50KiB



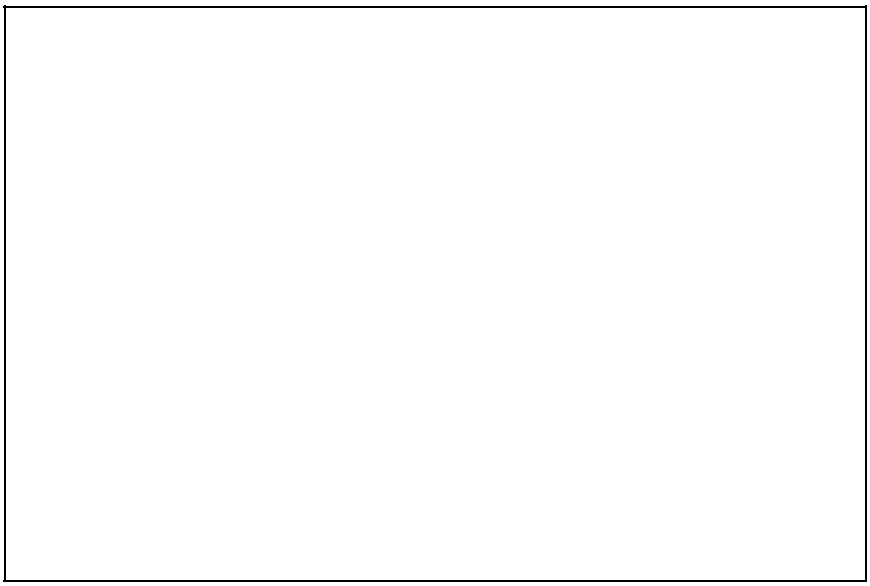
Reads/minio: one-node/100MiB



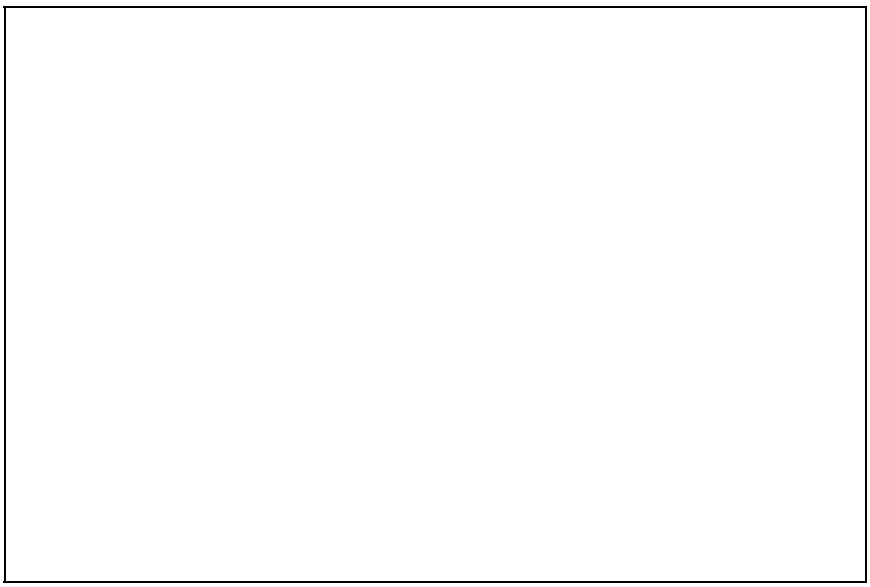
Reads/minio: one-node/1KiB



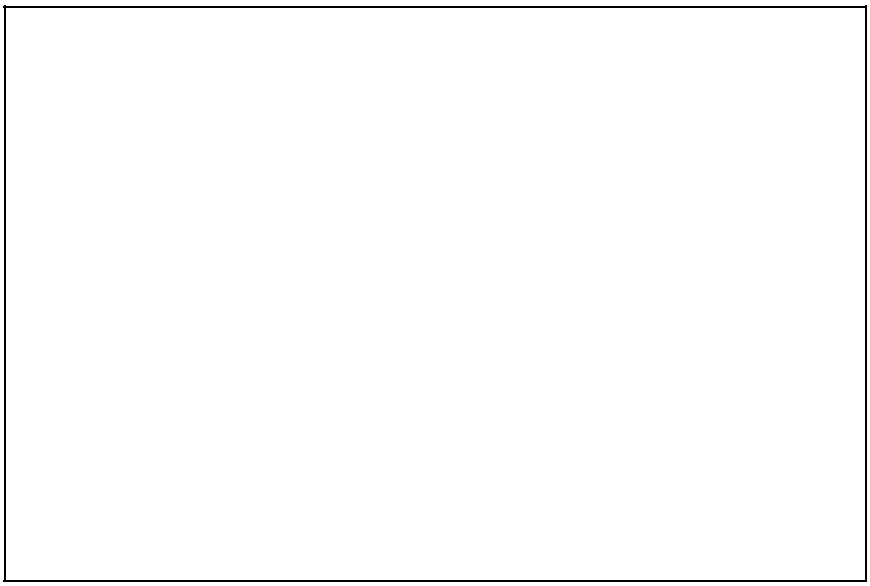
Reads/minio: one-node/1MiB



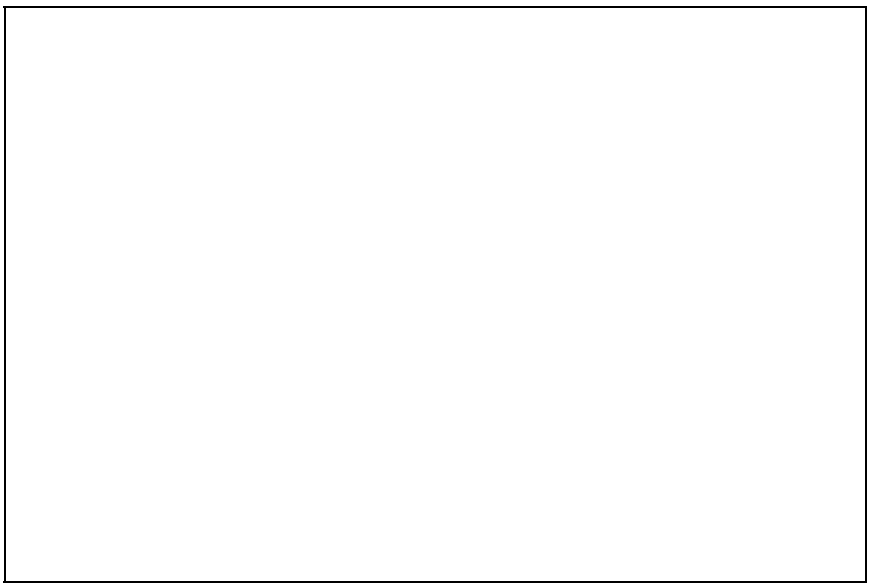
Reads/minio: one-node/50KiB



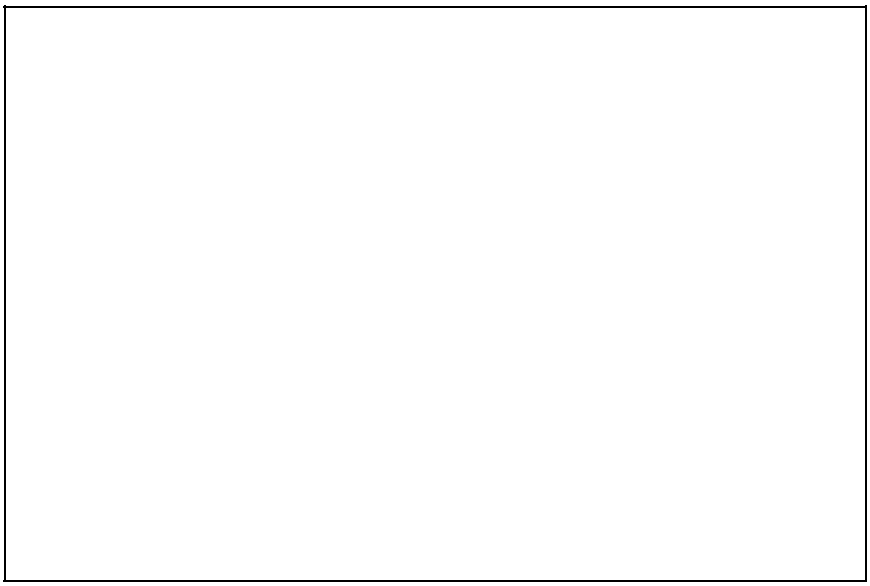
Reads/minio: two-node/100MiB



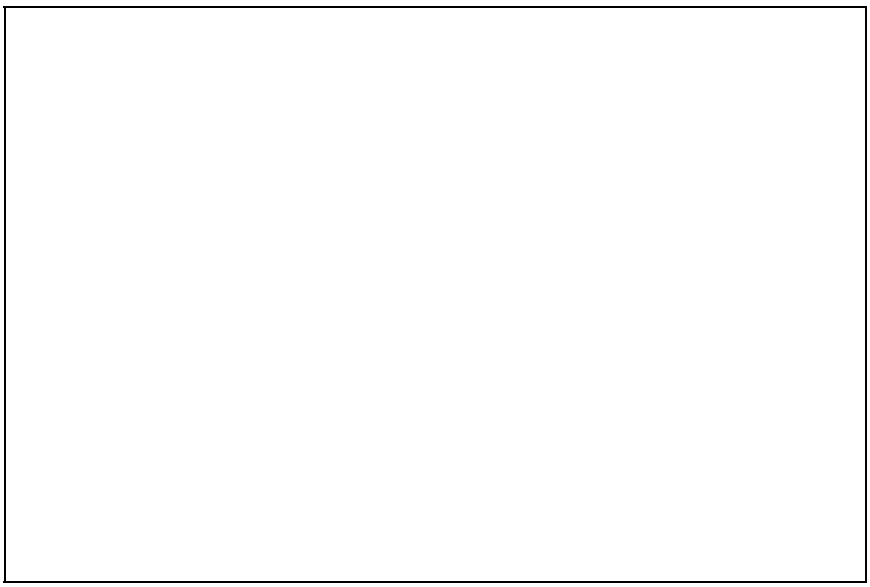
Reads/minio: two-node/1KiB

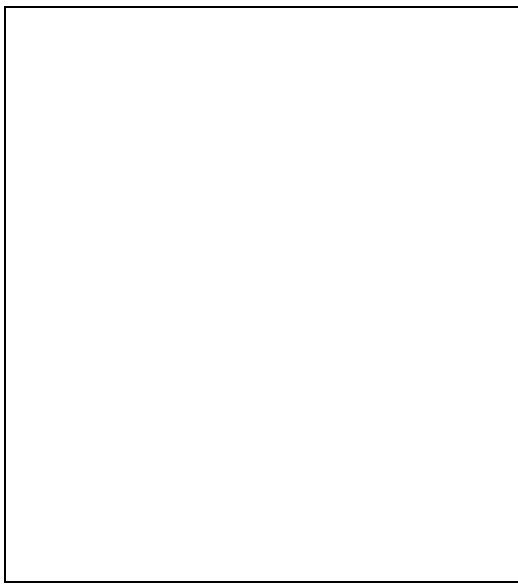


Reads/minio: two-node/1MiB



Reads/minio: two-node/50KiB





This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

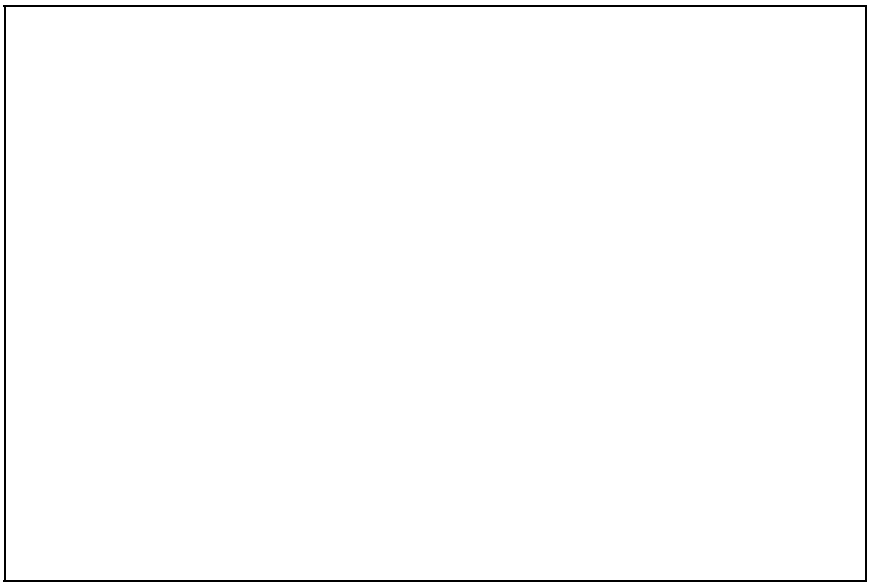
Writes/1.05GB

Violin Plot

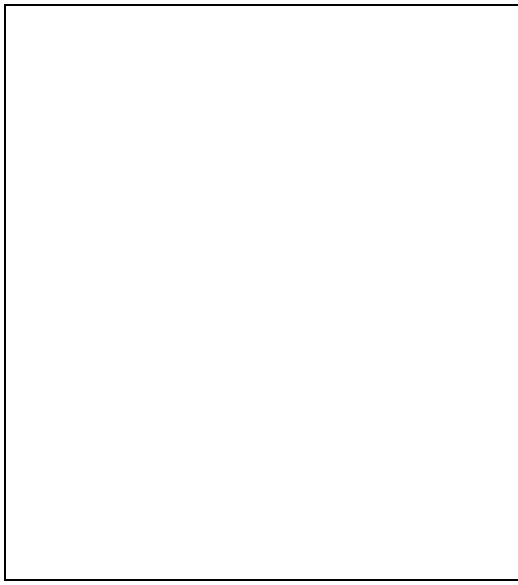
This chart shows the relationship between function/parameter and iteration time. The thickness of the shaded region indicates the probability that a measurement of the given function/parameter would take a particular length of time.

Writes/bob: one-node/1.05GB

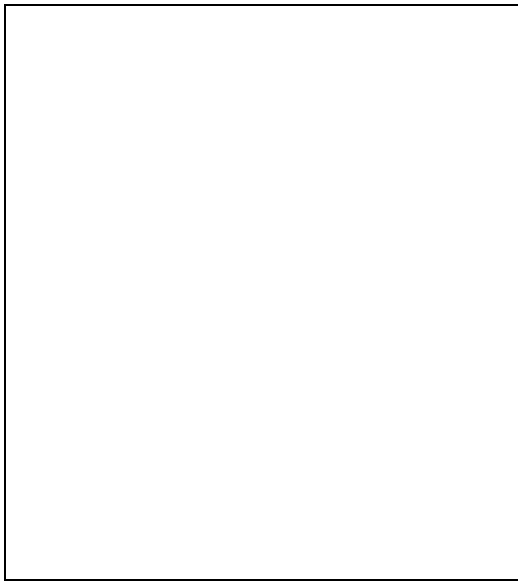
Writes/minio: one-node/1.05GB



Writes/bob: two-node/1.05GB



Writes/minio: two-node/1.05GB



This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

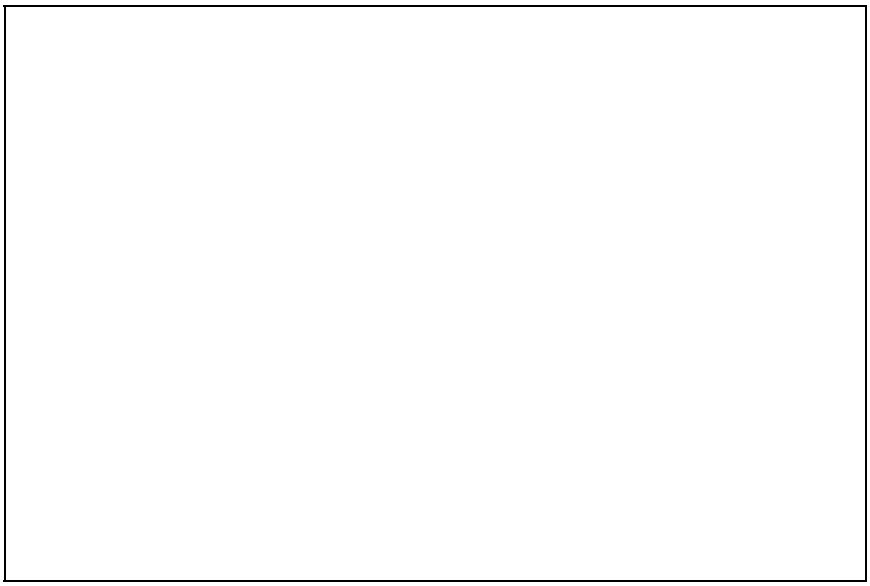
Writes/100MiB

Violin Plot

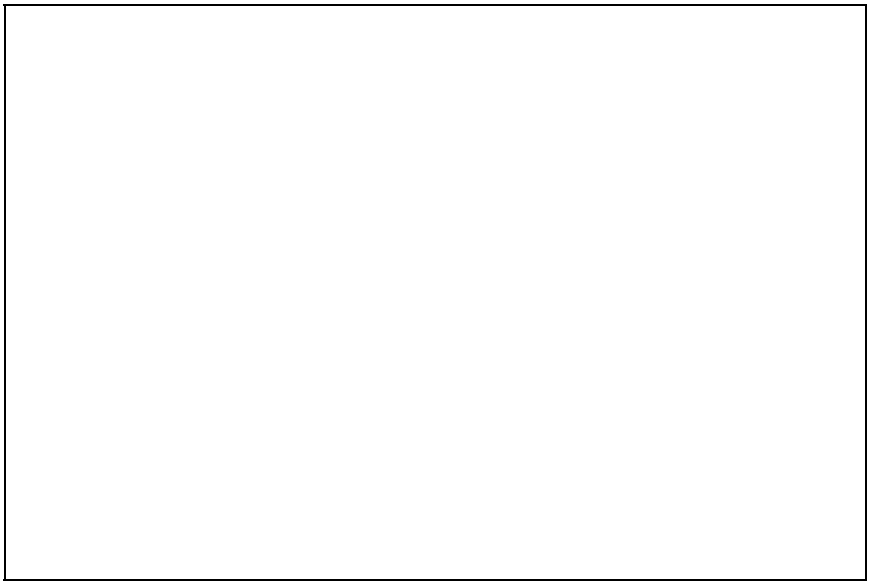
This chart shows the relationship between function/parameter and iteration time. The thickness of the shaded region indicates the probability that a measurement of the given function/parameter would take a particular length of time.

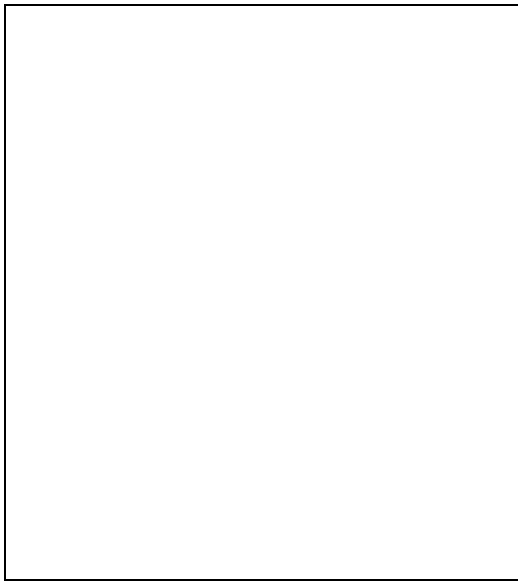
Writes/bob: one-node/100MiB

Writes/minio: one-node/100MiB

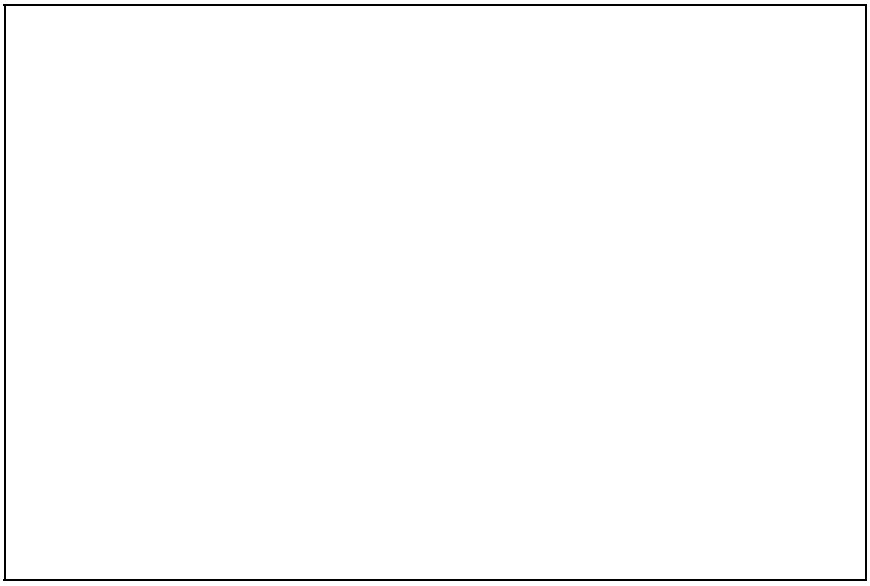


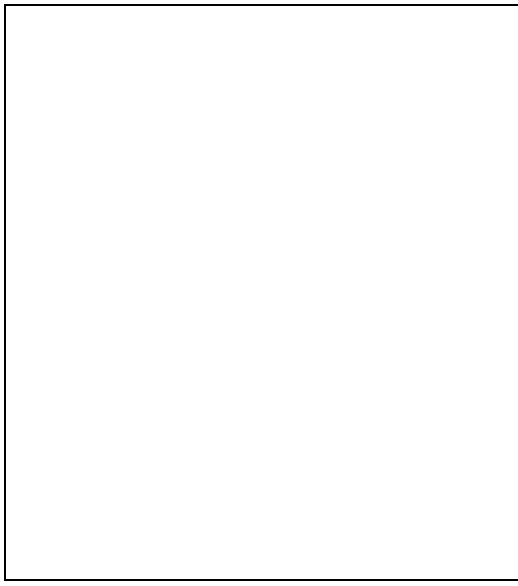
Writes/bob: two-node/100MiB





Writes/minio: two-node/100MiB





This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

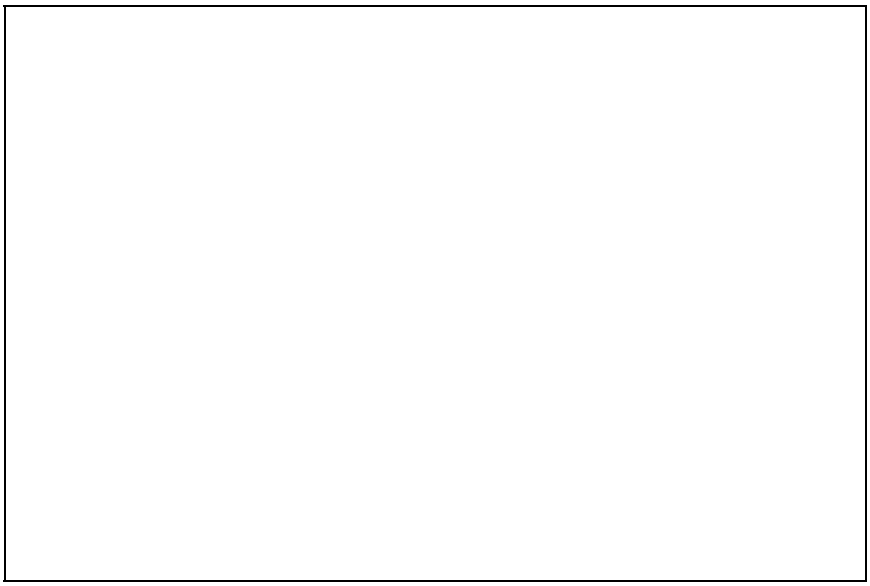
Writes/1KiB

Violin Plot

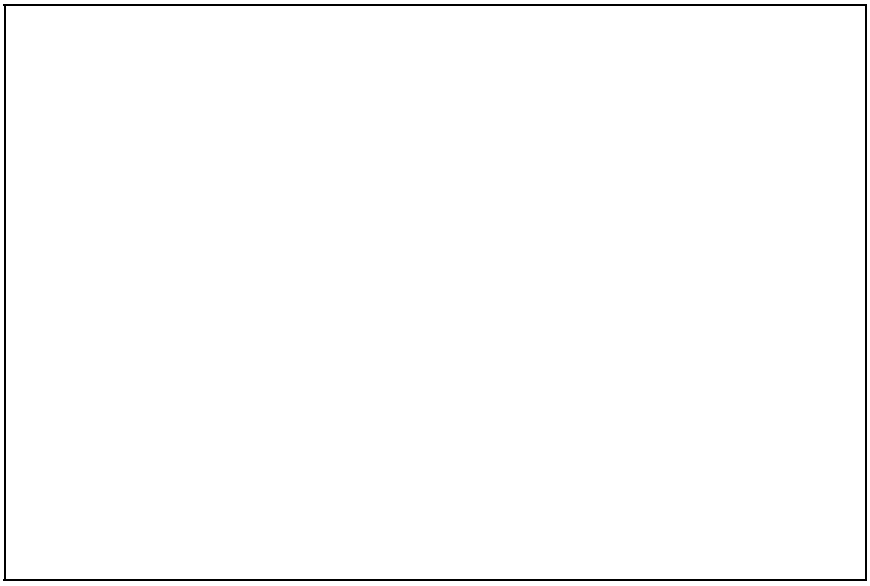
This chart shows the relationship between function/parameter and iteration time. The thickness of the shaded region indicates the probability that a measurement of the given function/parameter would take a particular length of time.

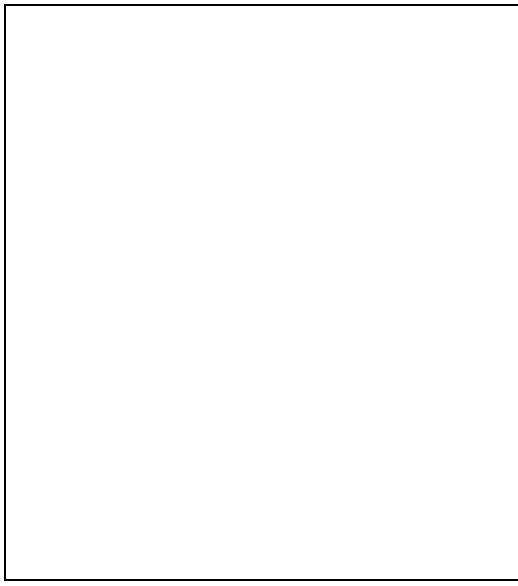
Writes/bob: one-node/1KiB

Writes/minio: one-node/1KiB

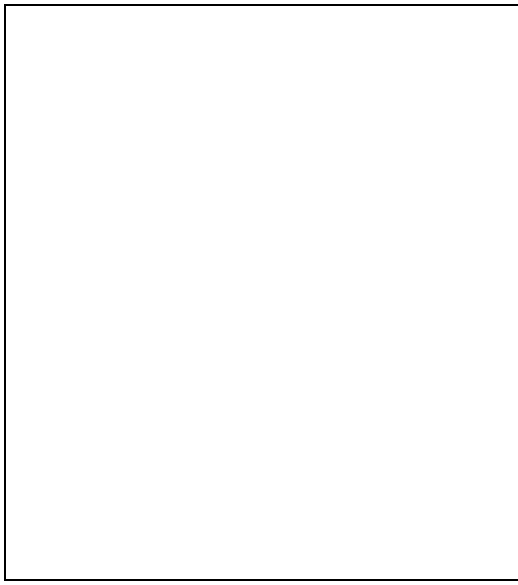


Writes/bob: two-node/1KiB





Writes/minio: two-node/1KiB



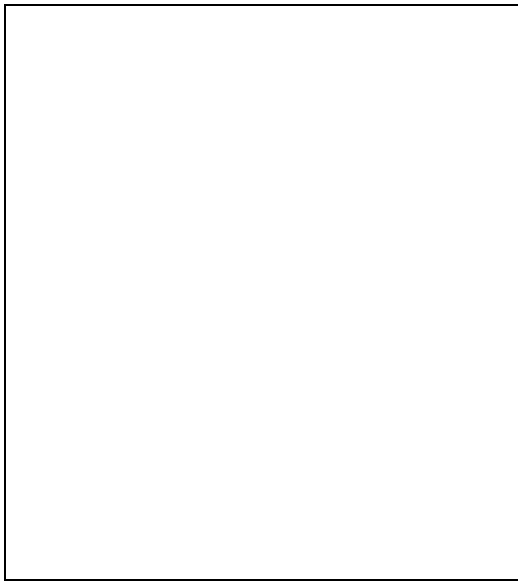
This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Writes/1MiB

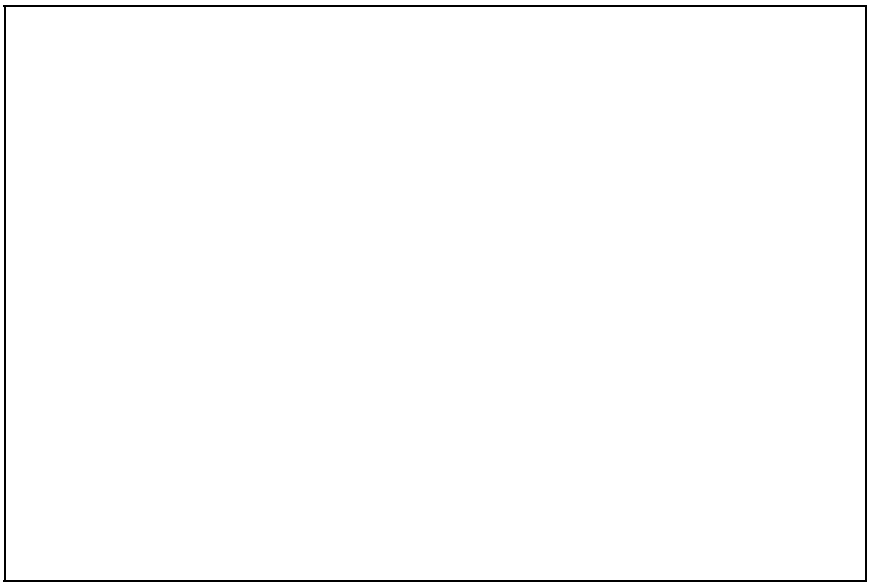
Violin Plot

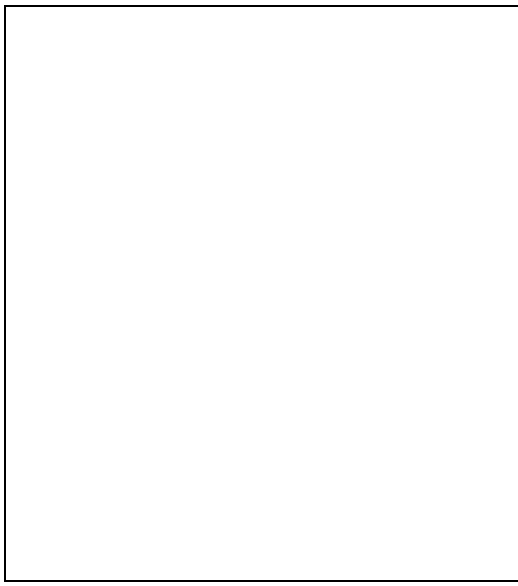
This chart shows the relationship between function/parameter and iteration time. The thickness of the shaded region indicates the probability that a measurement of the given function/parameter would take a particular length of time.

Writes/bob: one-node/1MiB

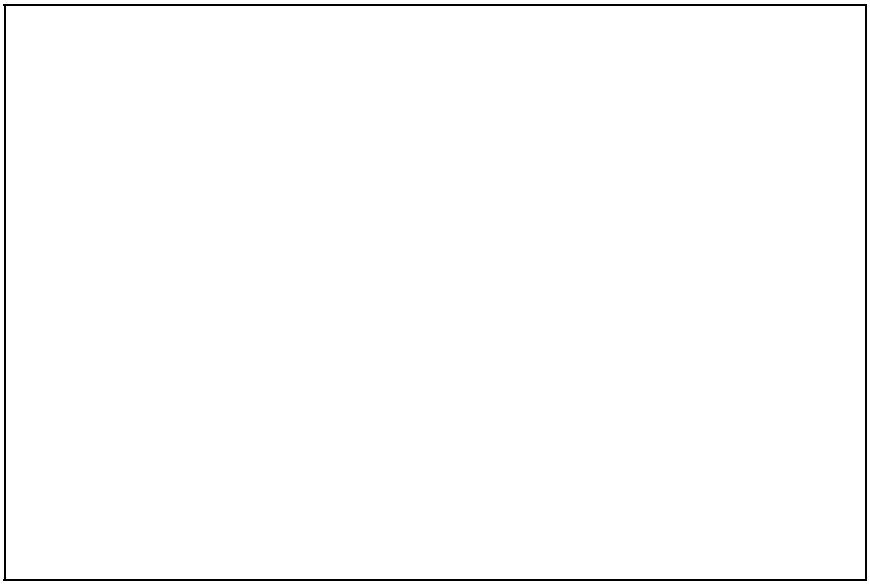


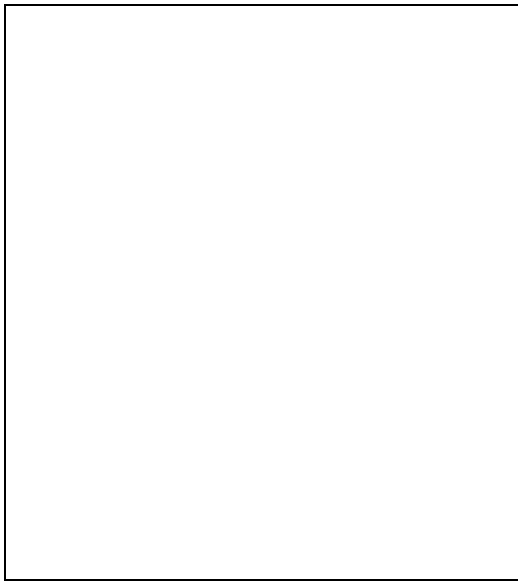
Writes/minio: one-node/1MiB



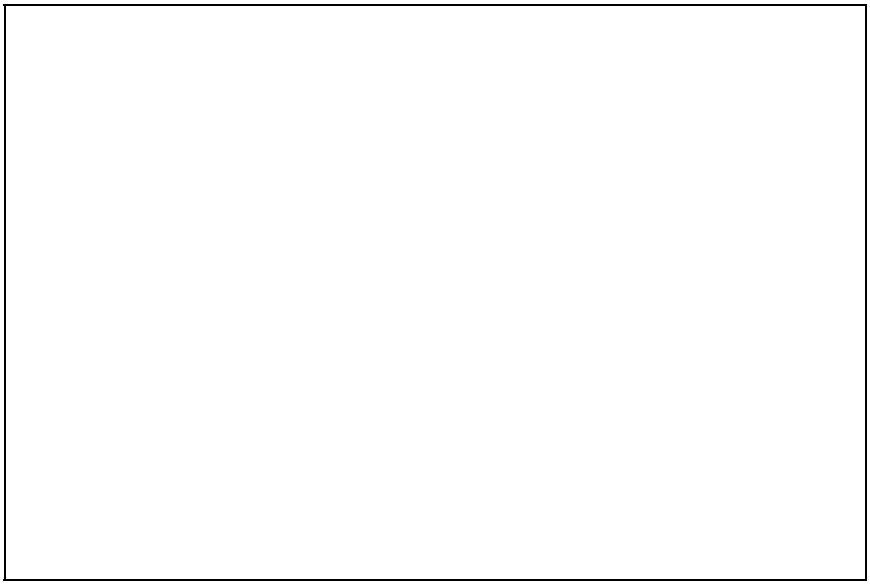


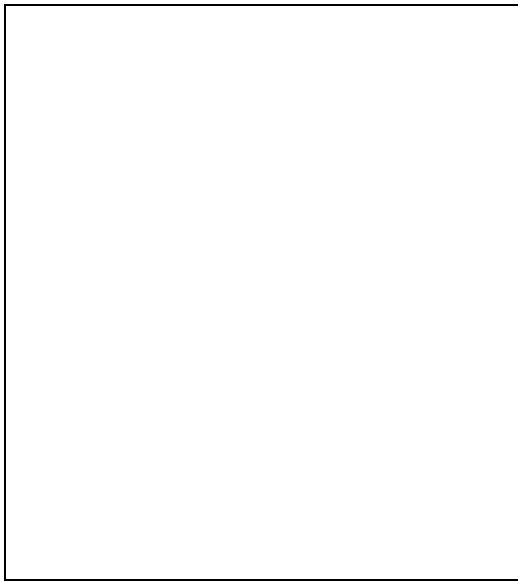
Writes/bob: two-node/1MiB





Writes/minio: two-node/1MiB





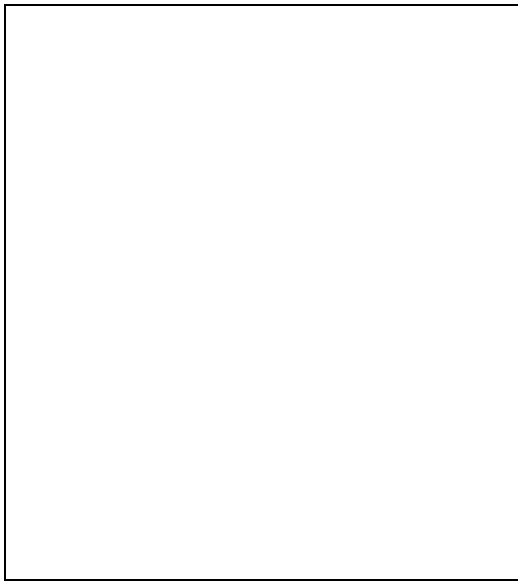
This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Writes/50KiB

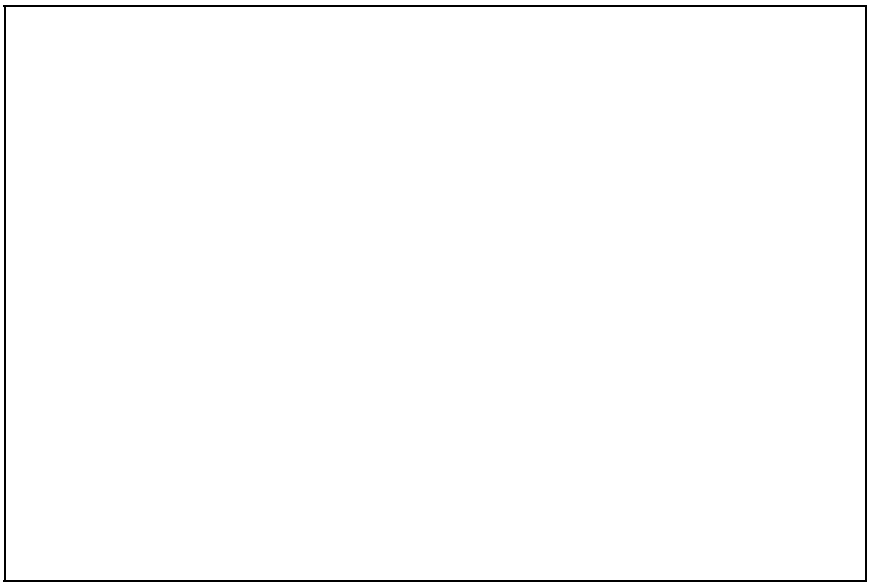
Violin Plot

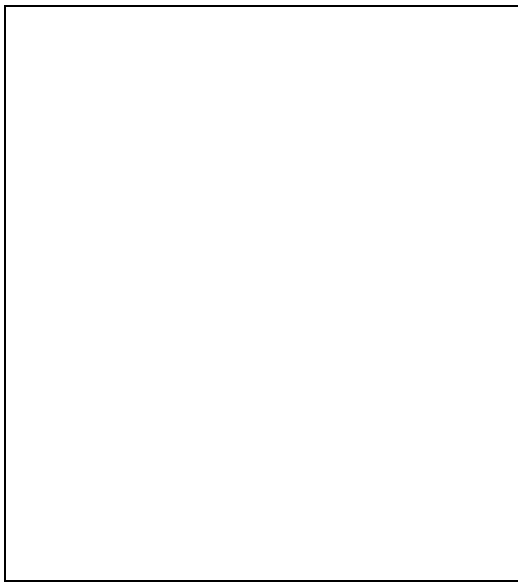
This chart shows the relationship between function/parameter and iteration time. The thickness of the shaded region indicates the probability that a measurement of the given function/parameter would take a particular length of time.

Writes/bob: one-node/50KiB

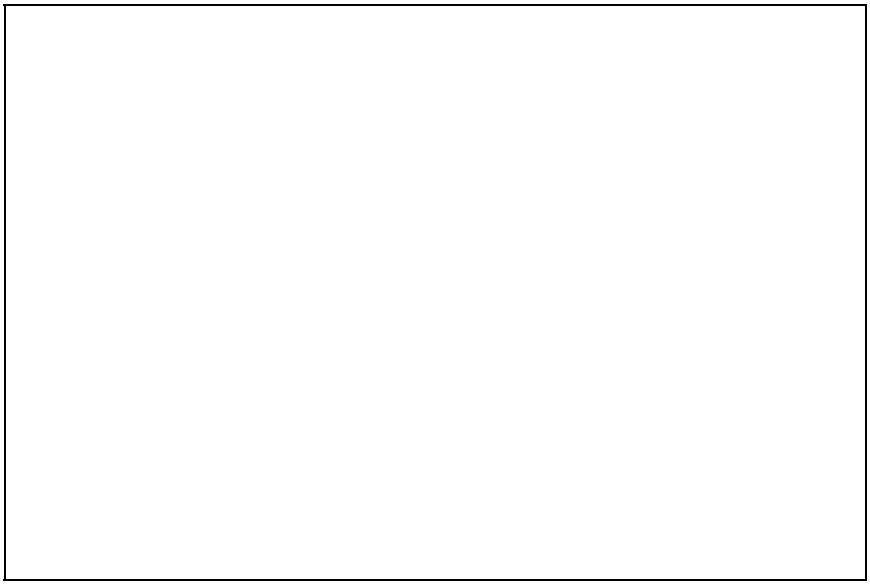


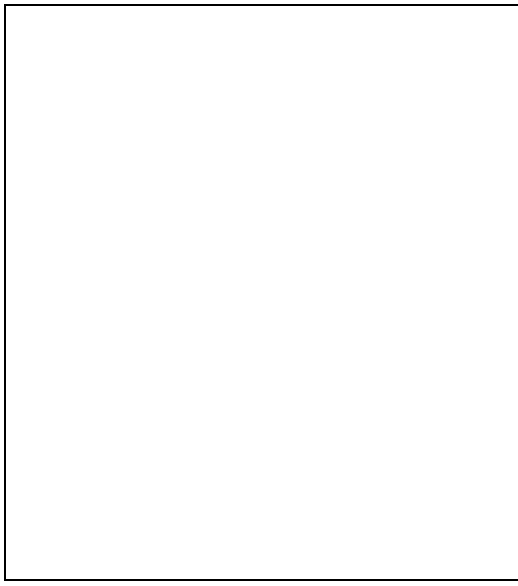
Writes/minio: one-node/50KiB



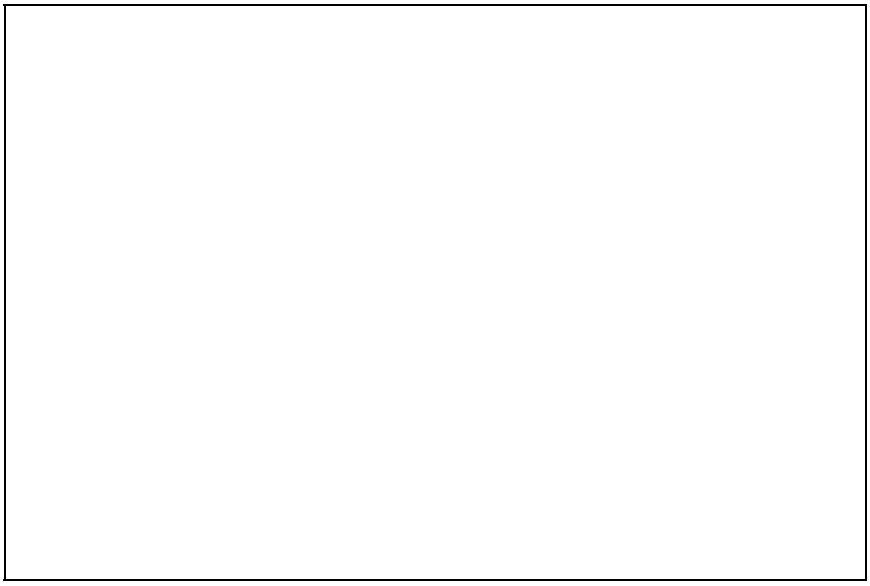


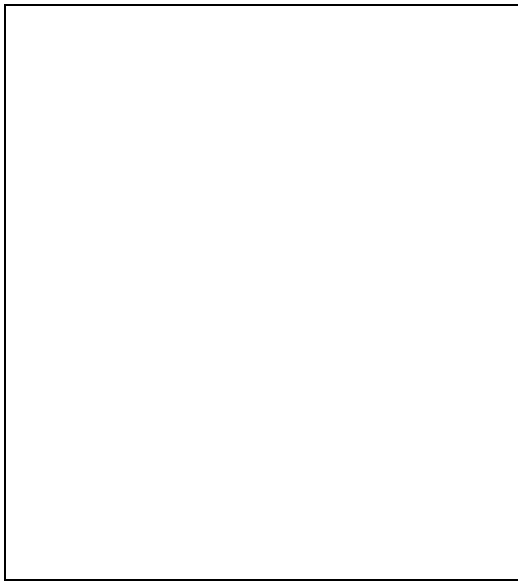
Writes/bob: two-node/50KiB





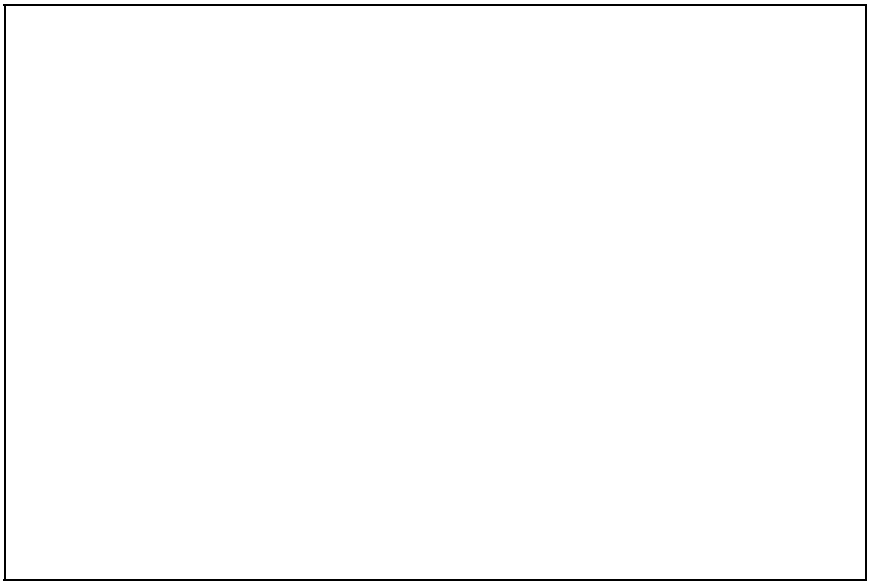
Writes/minio: two-node/50KiB

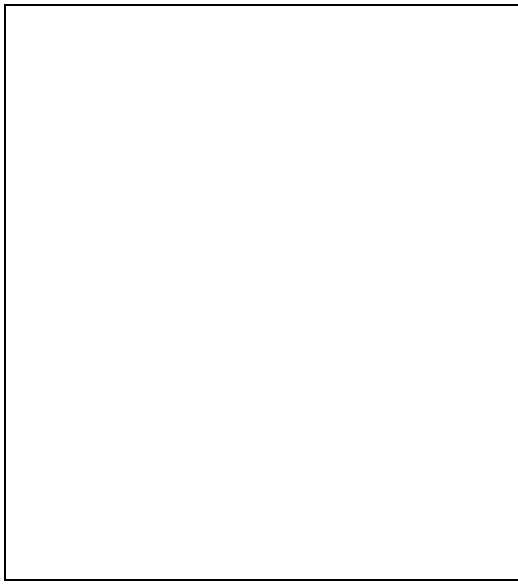




This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Writes/bob: one-node/1.05GB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Throughput	120.65 MiB/s	122.44 MiB/s	124.17 MiB/s
R²	0.0001632	0.0001691	0.0001624
Mean	8.0533 s	8.1670 s	8.2888 s
Std. Dev.	473.73 ms	602.52 ms	724.36 ms
Median	7.9596 s	8.0342 s	8.1475 s
MAD	312.21 ms	430.83 ms	593.84 ms

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD

Understanding this report:

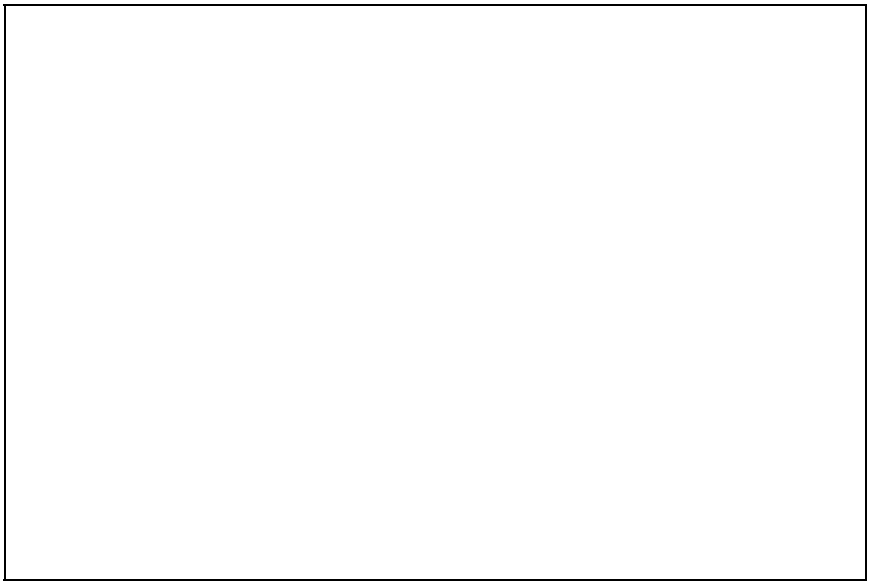
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

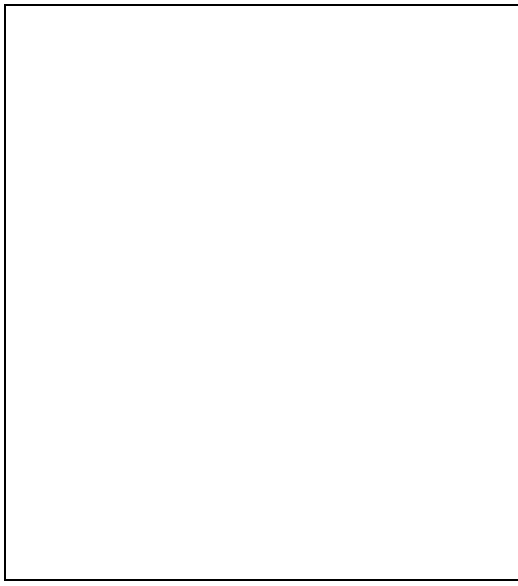
The plot on the right shows the average time per iteration for the samples. Each point represents one sample.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Writes/bob: one-node/100MiB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Throughput	62.566 MiB/s	66.959 MiB/s	71.561 MiB/s
R²	0.0051756	0.0053586	0.0051419
Mean	1.3974 s	1.4935 s	1.5983 s
Std. Dev.	390.04 ms	511.91 ms	618.45 ms
Median	1.2423 s	1.2913 s	1.4537 s
MAD	165.17 ms	266.46 ms	414.10 ms

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD

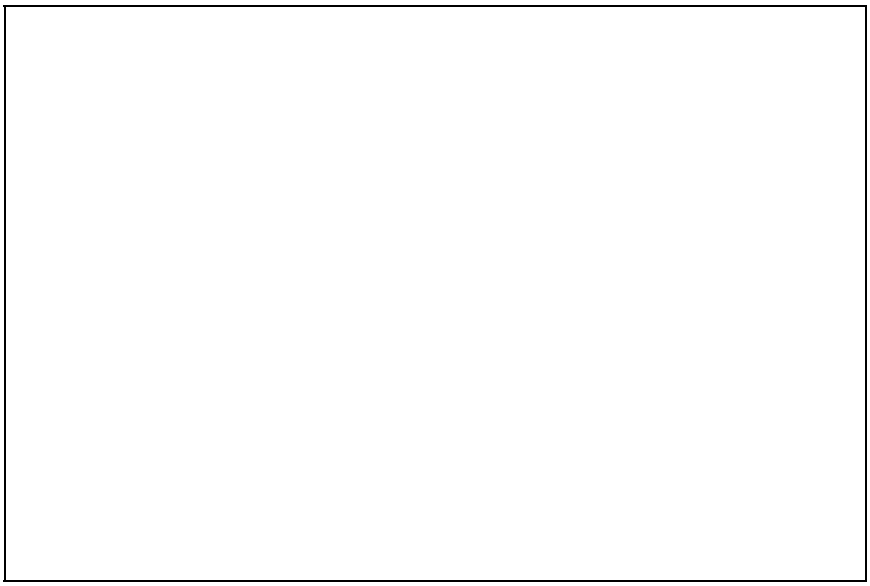
Understanding this report:

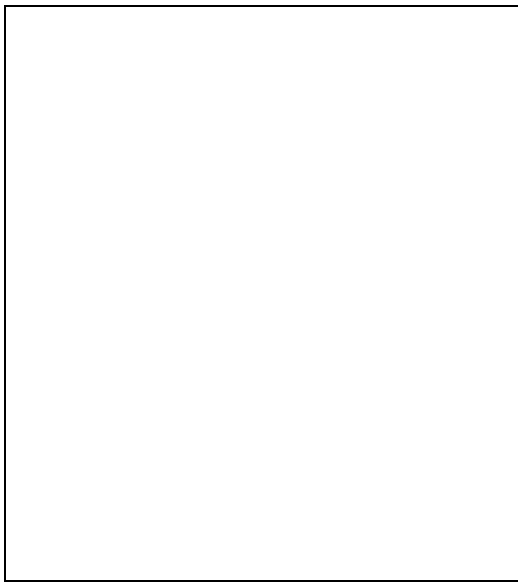
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the average time per iteration for the samples. Each point represents one sample.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	-2.5971%	+10.853%	+25.550%	(p = 0.10 > 0.05)
Change in throughput	+2.6663%	-9.7905%	-20.350%	

No change in performance detected.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

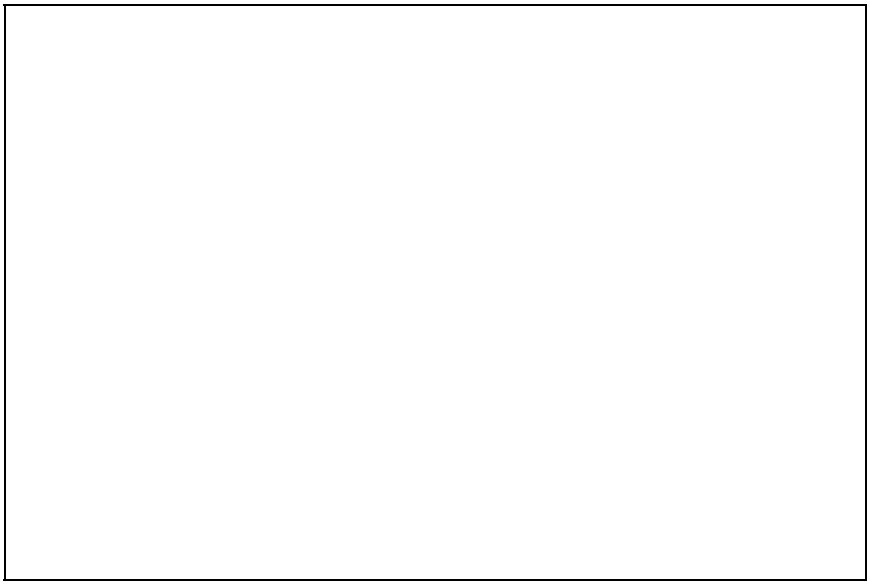
The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

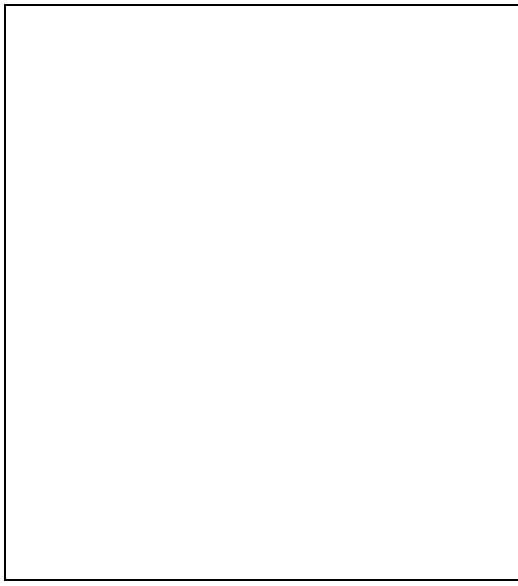
The plot on the right shows the iteration times for the two measurements. Again, the red dots represent the previous measurement while the blue dots show the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Writes/bob: one-node/1KiB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Slope	269.73 Âµs	274.32 Âµs	278.93 Âµs
Throughput	3.5011 MiB/s	3.5600 MiB/s	3.6206 MiB/s
R²	0.6877940	0.7015282	0.6876416
Mean	275.27 Âµs	279.21 Âµs	283.23 Âµs
Std. Dev.	17.681 Âµs	20.361 Âµs	22.714 Âµs
Median	270.78 Âµs	276.55 Âµs	283.18 Âµs
MAD	18.702 Âµs	25.507 Âµs	29.092 Âµs

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD
- Slope

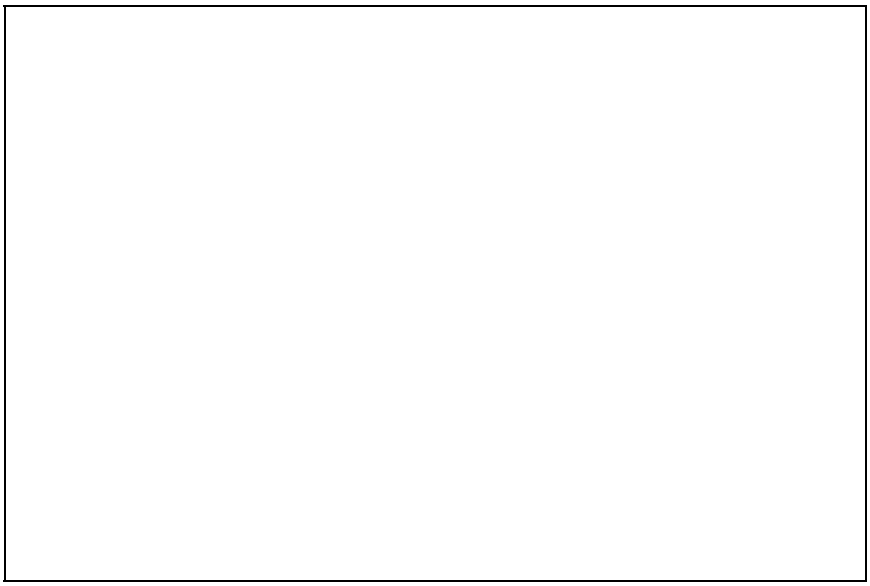
Understanding this report:

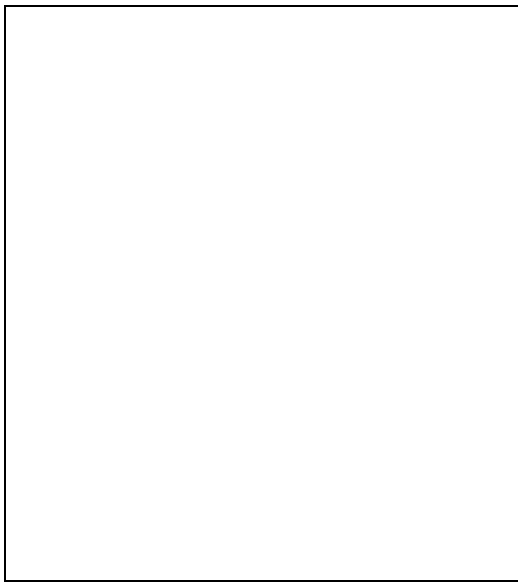
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the linear regression calculated from the measurements. Each point represents a sample, though here it shows the total time for the sample rather than time per iteration. The line is the line of best fit for these measurements.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	+7.5279%	+9.6863%	+11.818%	(p = 0.00 < 0.05)
Change in throughput	-7.0009%	-8.8309%	-10.569%	

Performance has regressed.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

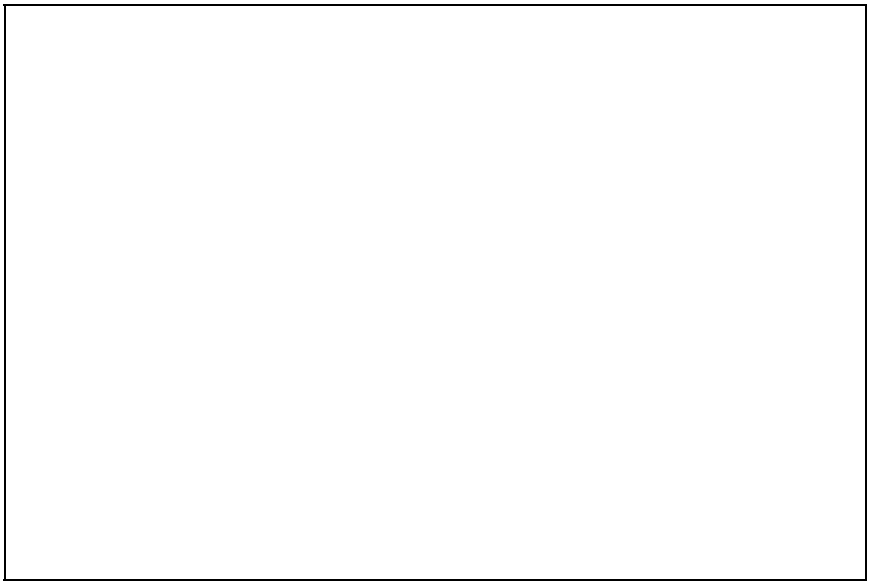
The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

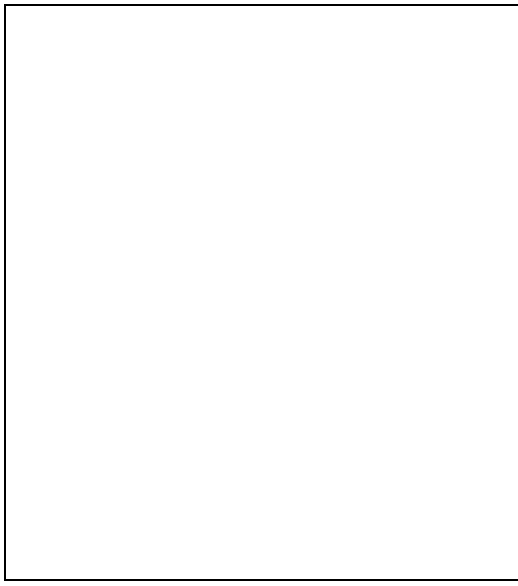
The plot on the right shows the two regressions. Again, the red line represents the previous measurement while the blue line shows the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Writes/bob: one-node/1MiB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Throughput	141.75 MiB/s	142.67 MiB/s	143.56 MiB/s
R²	0.0008774	0.0009094	0.0008747
Mean	6.9659 ms	7.0094 ms	7.0548 ms
Std. Dev.	193.11 Âµs	228.87 Âµs	261.95 Âµs
Median	6.9103 ms	6.9538 ms	7.0376 ms
MAD	176.45 Âµs	240.11 Âµs	291.15 Âµs

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD

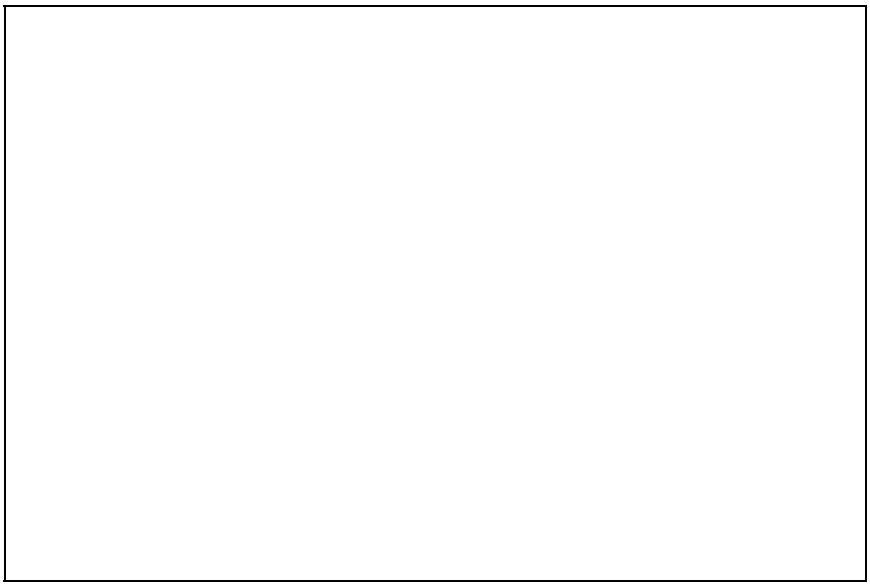
Understanding this report:

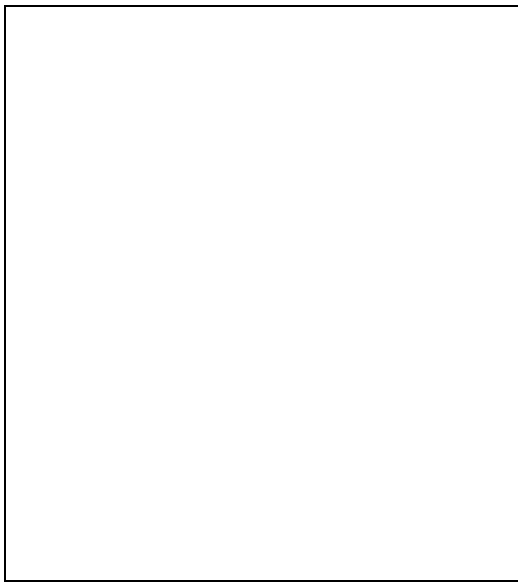
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the average time per iteration for the samples. Each point represents one sample.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	-37.544%	-19.560%	-4.1898%	(p = 0.07 > 0.05)
Change in throughput	+60.113%	+24.316%	+4.3730%	

No change in performance detected.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

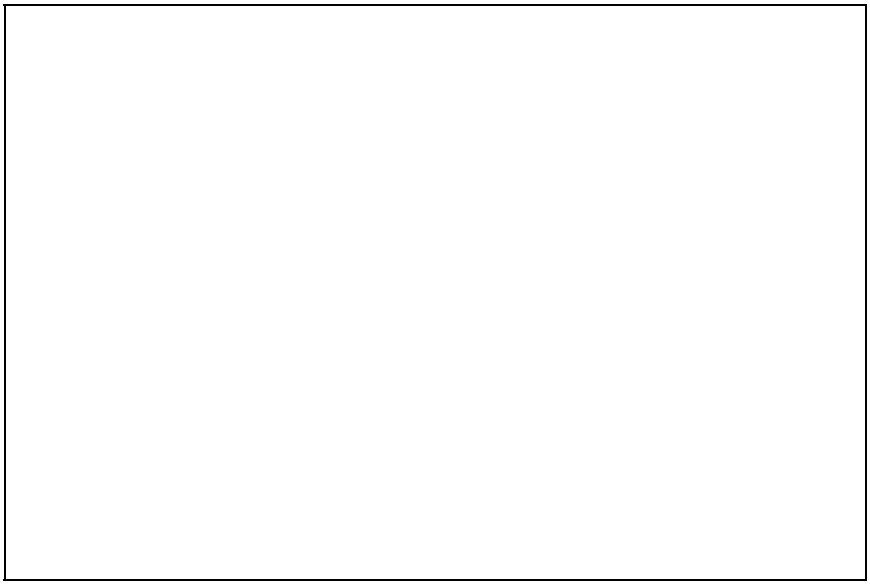
The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

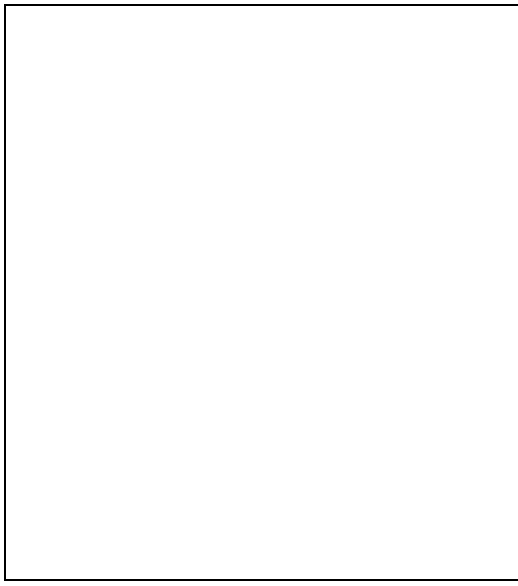
The plot on the right shows the iteration times for the two measurements. Again, the red dots represent the previous measurement while the blue dots show the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Writes/bob: one-node/50KiB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Slope	670.83 Âµs	677.43 Âµs	685.60 Âµs
Throughput	71.220 MiB/s	72.079 MiB/s	72.787 MiB/s
R²	0.7646247	0.7725927	0.7604126
Mean	665.63 Âµs	670.77 Âµs	676.79 Âµs
Std. Dev.	16.764 Âµs	28.597 Âµs	40.783 Âµs
Median	662.79 Âµs	664.46 Âµs	669.06 Âµs
MAD	9.4687 Âµs	12.625 Âµs	18.396 Âµs

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD
- Slope

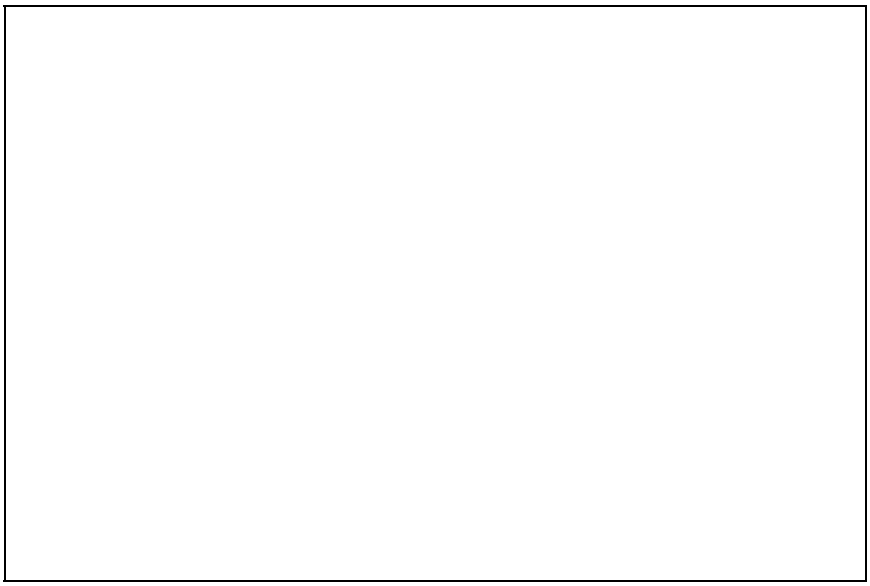
Understanding this report:

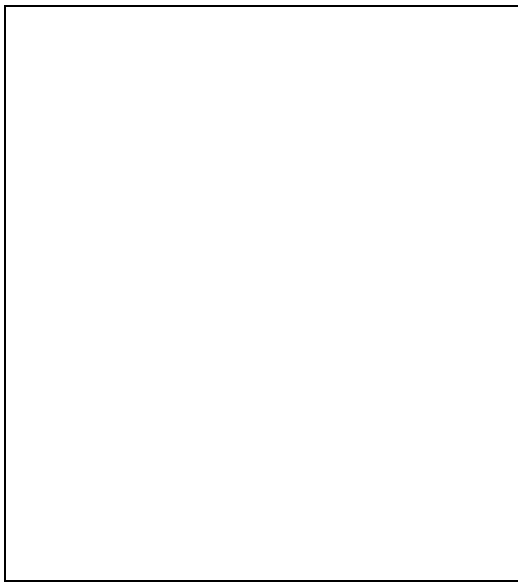
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the linear regression calculated from the measurements. Each point represents a sample, though here it shows the total time for the sample rather than time per iteration. The line is the line of best fit for these measurements.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	+0.3135%	+1.8336%	+3.4893%	(p = 0.02 < 0.05)
Change in throughput	-0.3125%	-1.8006%	-3.3716%	

Change within noise threshold.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

The plot on the right shows the two regressions. Again, the red line represents the previous measurement while the blue line shows the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Writes/bob: one-node

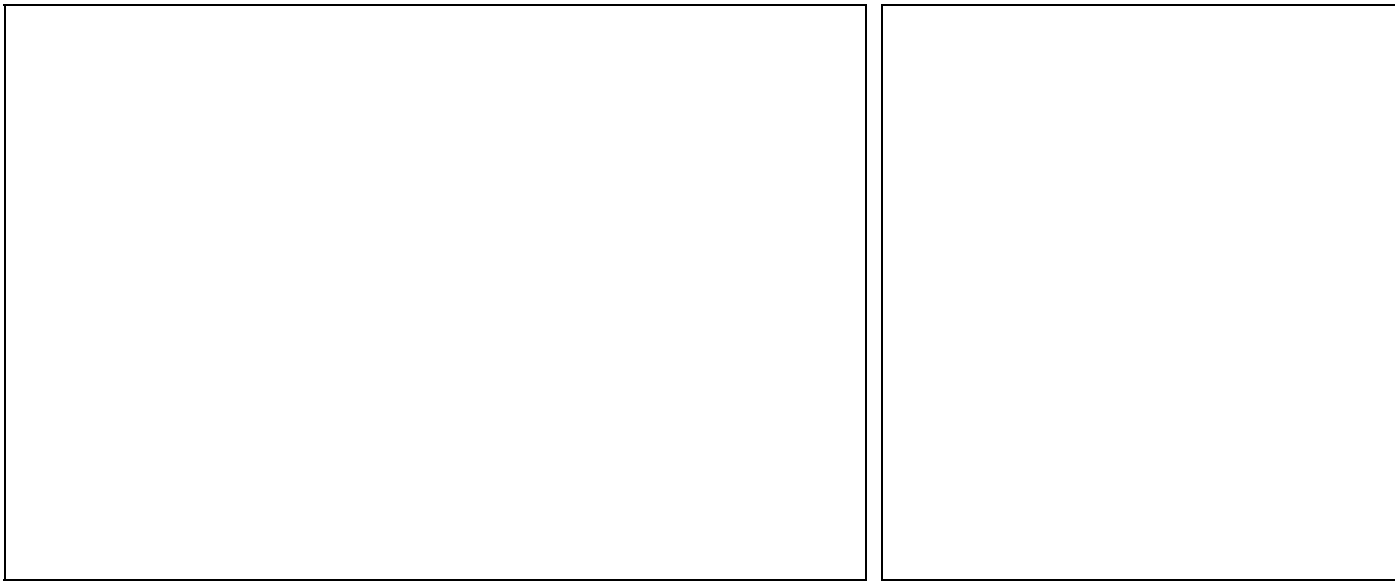
Violin Plot

This chart shows the relationship between function/parameter and iteration time. The thickness of the shaded region indicates the probability that a measurement of the given function/parameter would take a particular length of time.

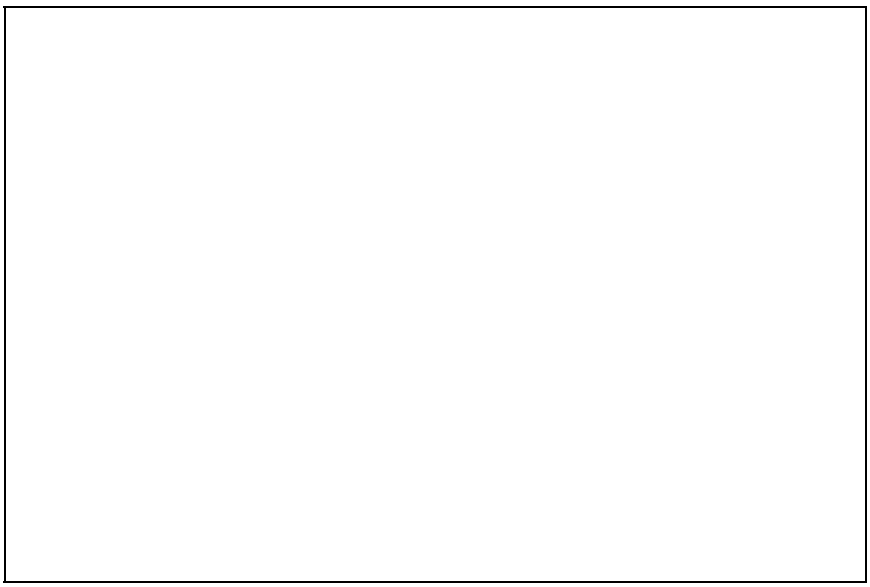
Line Chart

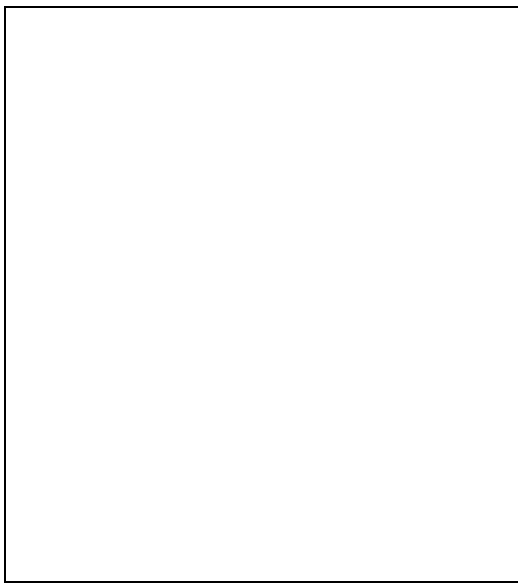
This chart shows the mean measured time for each function as the input (or the size of the input) increases.

Writes/bob: one-node/1KiB

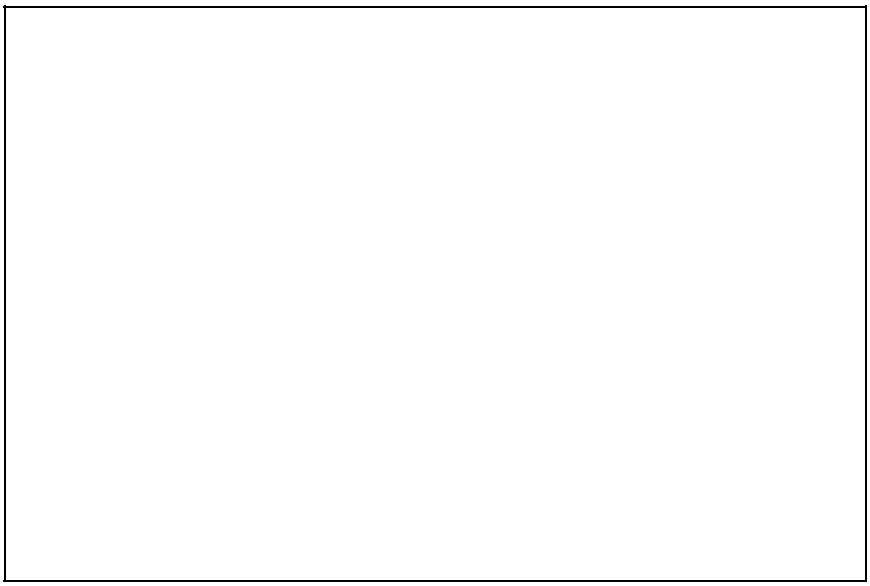


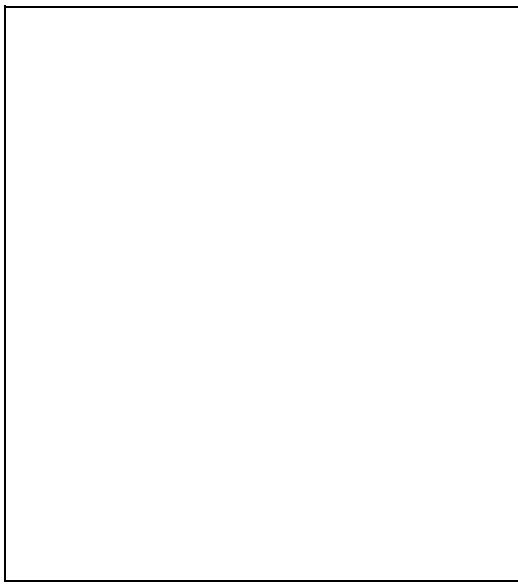
Writes/bob: one-node/50KiB



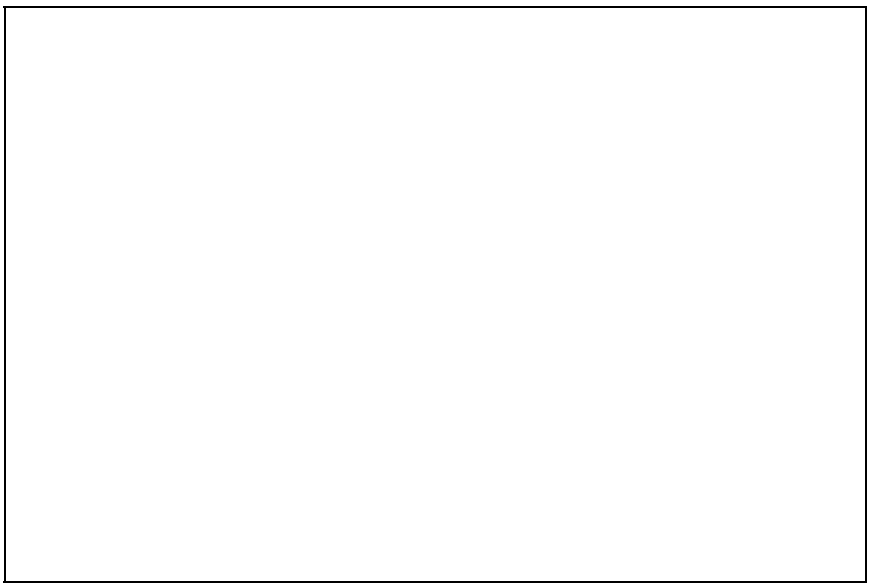


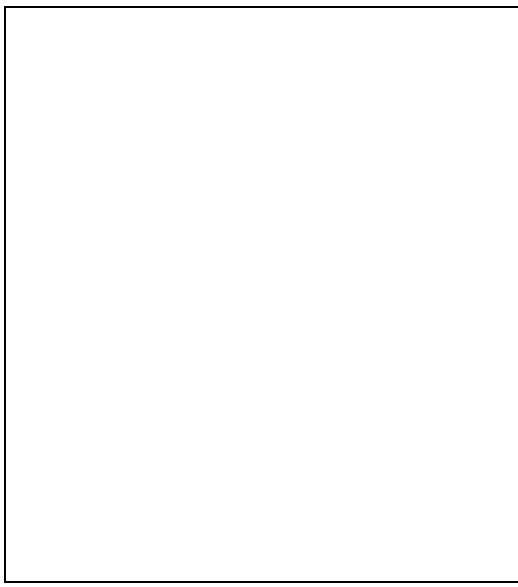
Writes/bob: one-node/1MiB





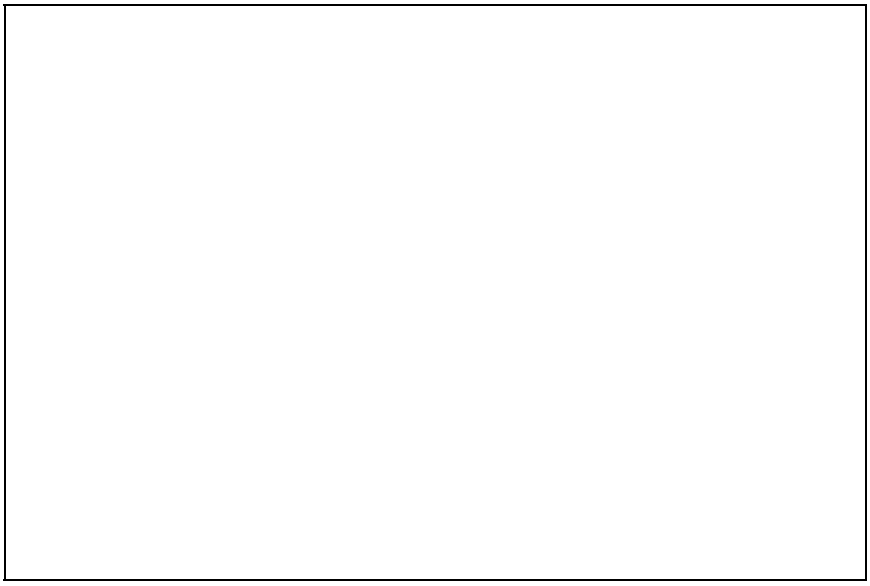
Writes/bob: one-node/100MiB

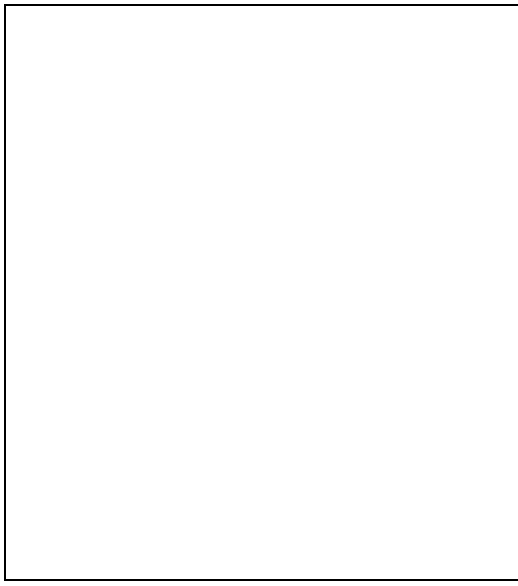




This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Writes/bob: two-node/1.05GB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Throughput	65.752 MiB/s	66.543 MiB/s	67.307 MiB/s
R^2	0.0240237	0.0248748	0.0239194
Mean	14.857 s	15.028 s	15.209 s
Std. Dev.	735.16 ms	898.36 ms	1.0496 s
Median	14.683 s	14.762 s	14.900 s
MAD	481.48 ms	678.63 ms	869.69 ms

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD

Understanding this report:

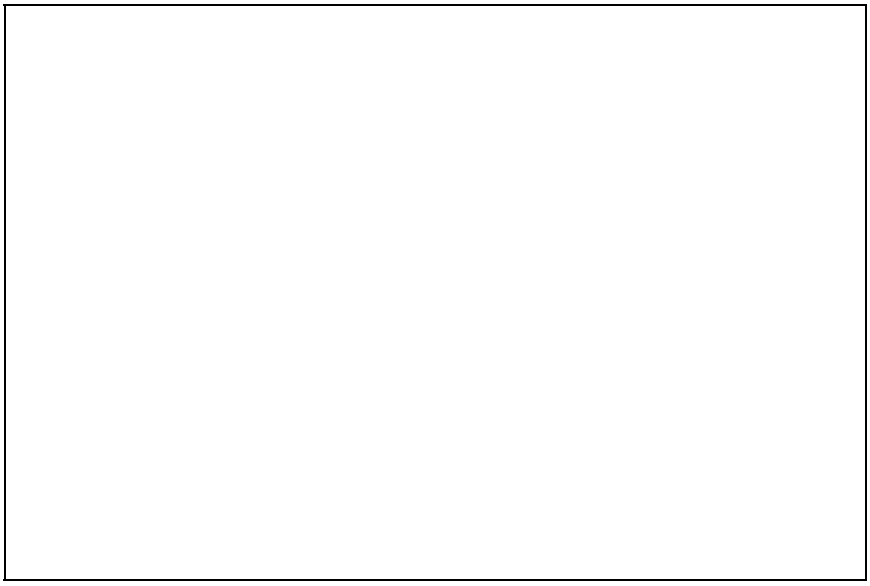
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

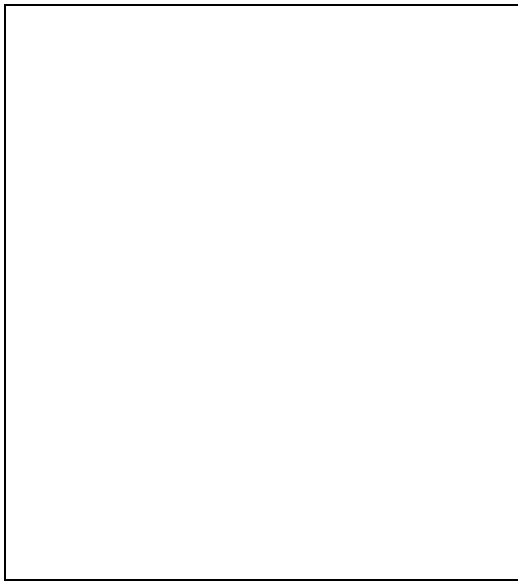
The plot on the right shows the average time per iteration for the samples. Each point represents one sample.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Writes/bob: two-node/100MiB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Throughput	34.949 MiB/s	36.680 MiB/s	38.522 MiB/s
R^2	0.0053865	0.0055857	0.0053726
Mean	2.5959 s	2.7263 s	2.8613 s
Std. Dev.	542.93 ms	679.38 ms	814.23 ms
Median	2.6104 s	2.6639 s	2.7771 s
MAD	313.54 ms	436.66 ms	787.93 ms

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD

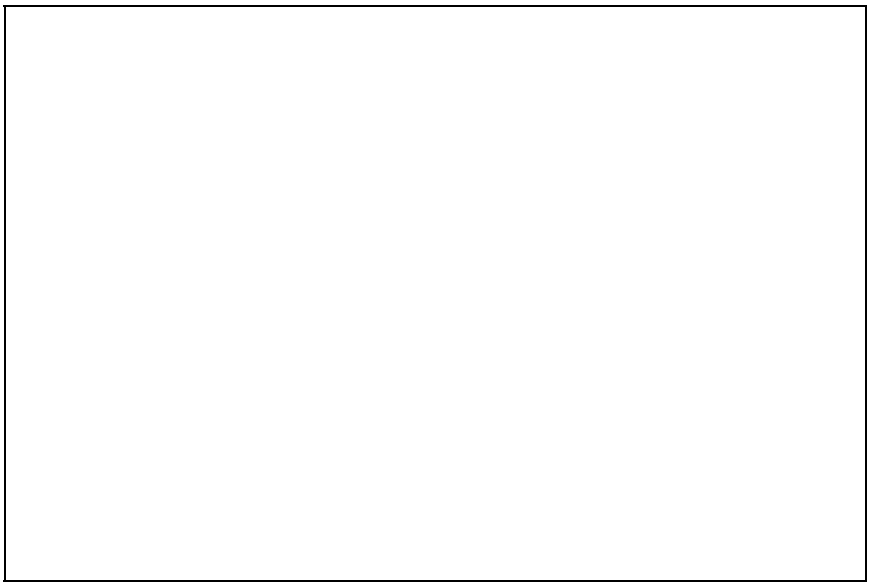
Understanding this report:

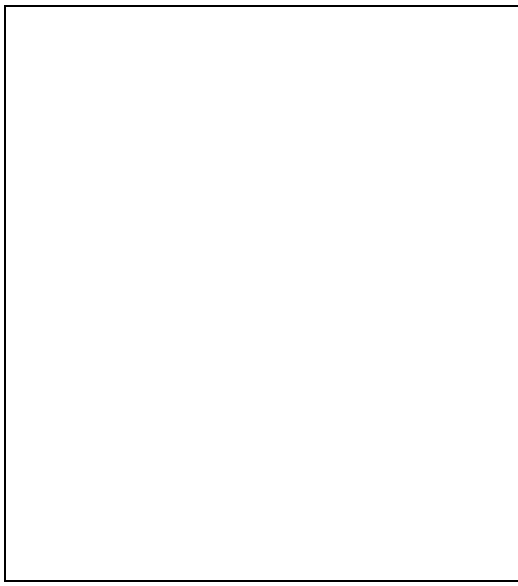
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the average time per iteration for the samples. Each point represents one sample.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	+22.349%	+32.011%	+42.490%	(p = 0.00 < 0.05)
Change in throughput	-18.266%	-24.249%	-29.820%	

Performance has regressed.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

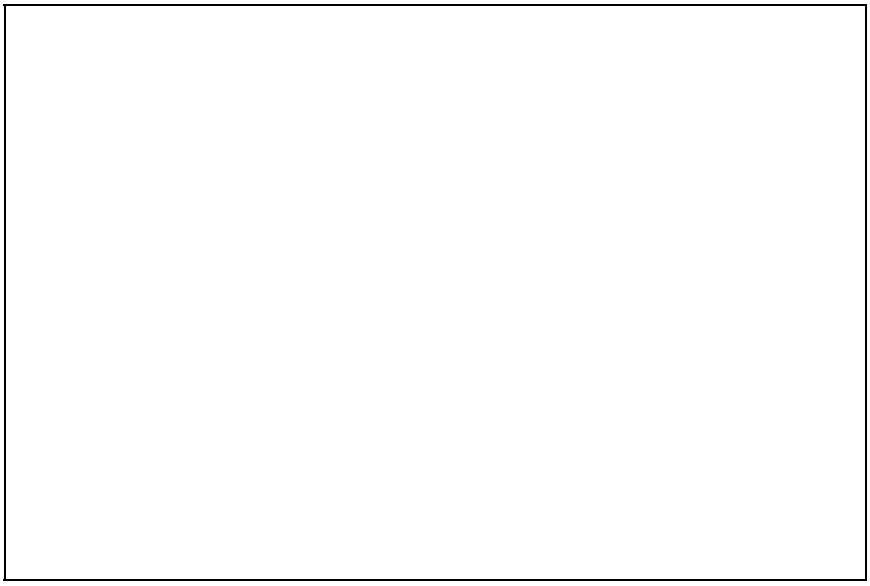
The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

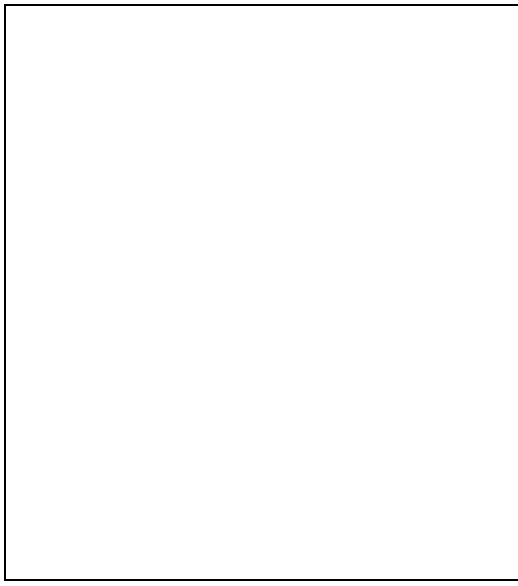
The plot on the right shows the iteration times for the two measurements. Again, the red dots represent the previous measurement while the blue dots show the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Writes/bob: two-node/1KiB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Slope	635.41 Âµs	640.78 Âµs	646.53 Âµs
Throughput	1.5105 MiB/s	1.5240 MiB/s	1.5369 MiB/s
R²	0.8138633	0.8220455	0.8126580
Mean	631.08 Âµs	636.36 Âµs	641.83 Âµs
Std. Dev.	20.416 Âµs	27.457 Âµs	33.899 Âµs
Median	634.21 Âµs	637.05 Âµs	639.85 Âµs
MAD	10.256 Âµs	13.472 Âµs	18.461 Âµs

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD
- Slope

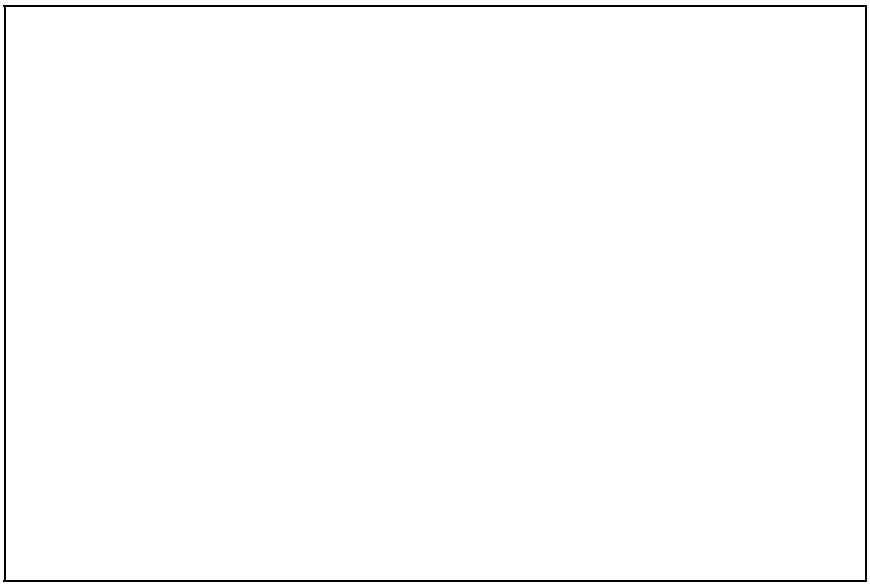
Understanding this report:

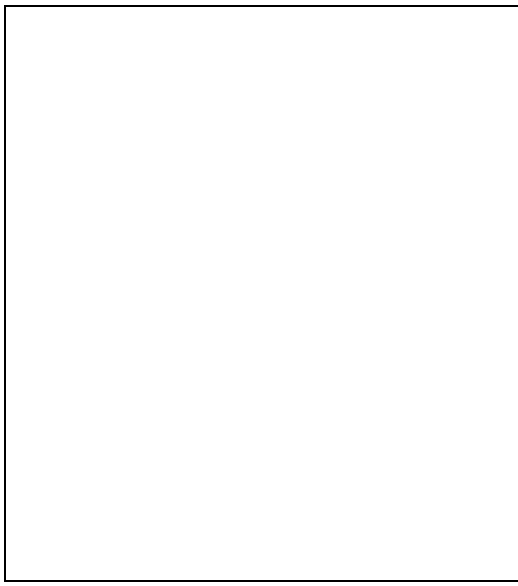
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the linear regression calculated from the measurements. Each point represents a sample, though here it shows the total time for the sample rather than time per iteration. The line is the line of best fit for these measurements.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	-6.4443%	-5.0735%	-3.8252%	(p = 0.00 < 0.05)
Change in throughput	+6.8882%	+5.3447%	+3.9773%	

Performance has improved.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

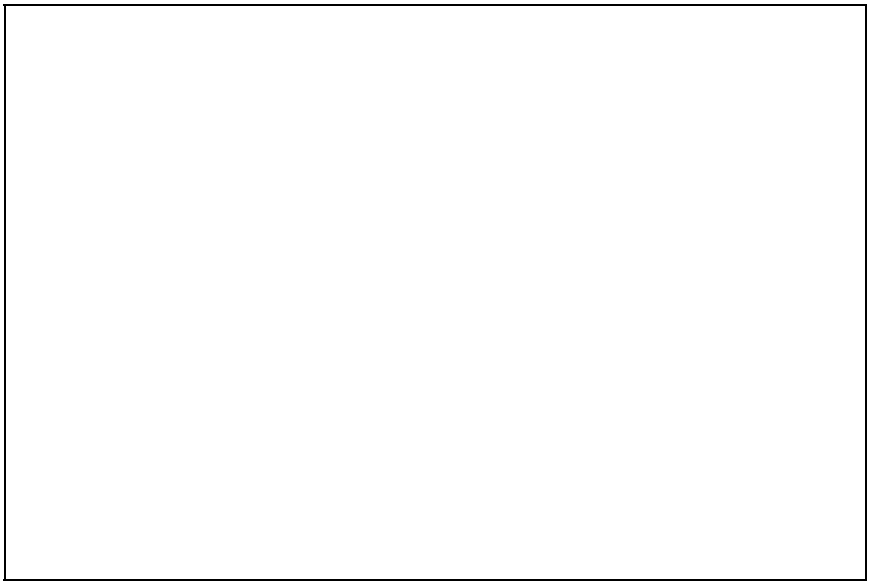
The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

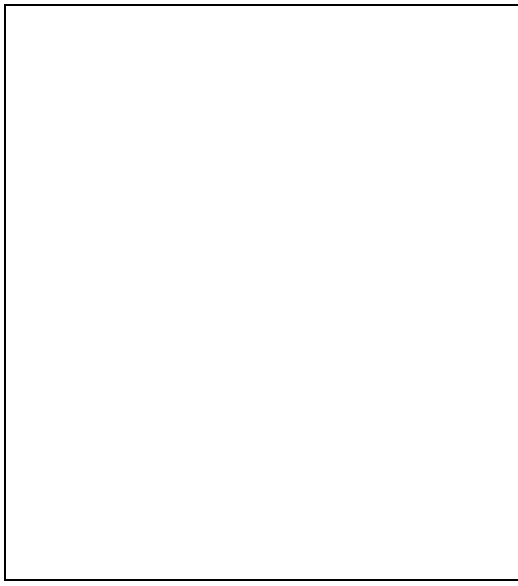
The plot on the right shows the two regressions. Again, the red line represents the previous measurement while the blue line shows the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Writes/bob: two-node/1MiB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Throughput	63.952 MiB/s	64.680 MiB/s	65.413 MiB/s
R²	0.0000989	0.0001026	0.0000988
Mean	15.287 ms	15.461 ms	15.637 ms
Std. Dev.	742.85 Âµs	891.64 Âµs	1.0311 ms
Median	15.258 ms	15.516 ms	15.748 ms
MAD	670.83 Âµs	864.70 Âµs	1.0169 ms

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD

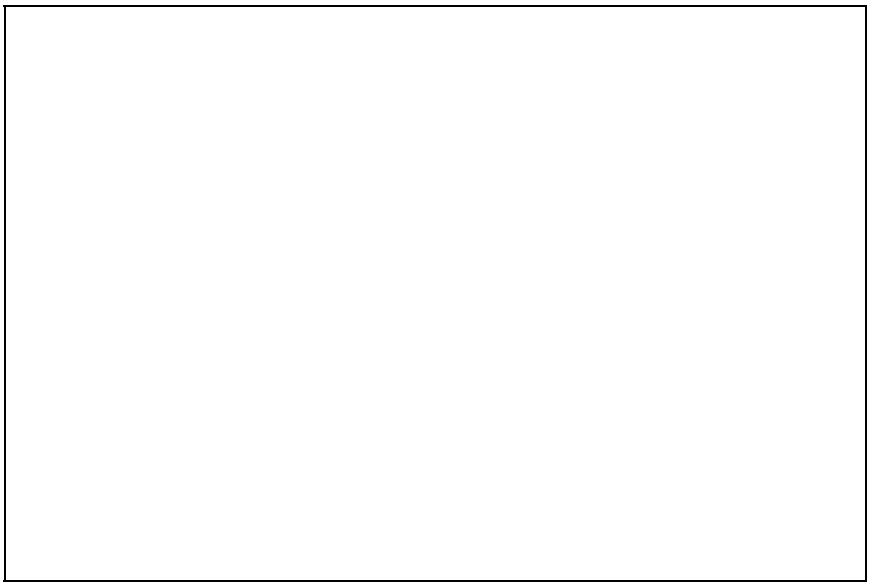
Understanding this report:

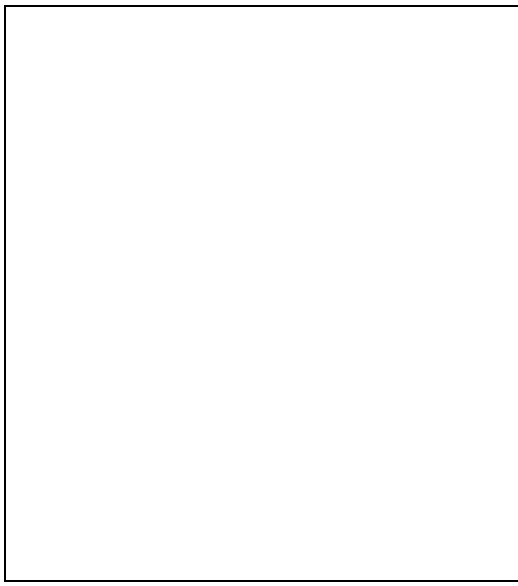
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the average time per iteration for the samples. Each point represents one sample.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	-5.0270%	-2.5047%	+0.1365%	(p = 0.06 > 0.05)
Change in throughput	+5.2931%	+2.5691%	-0.1363%	

No change in performance detected.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

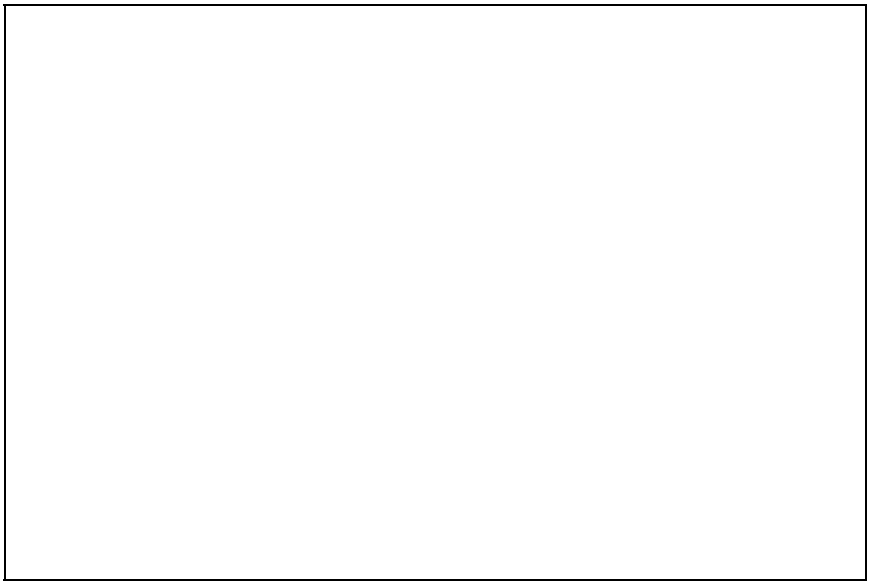
The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

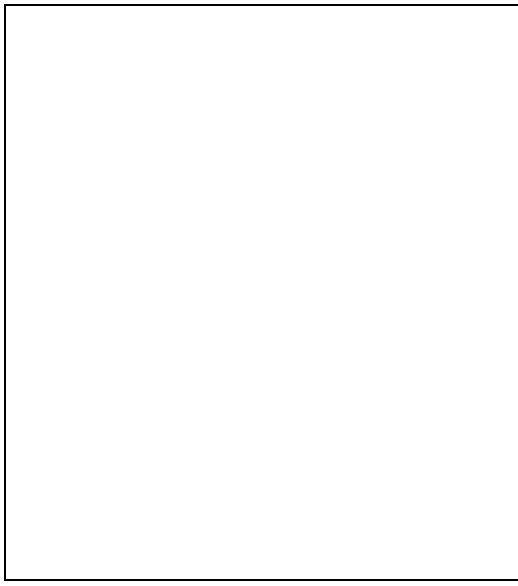
The plot on the right shows the iteration times for the two measurements. Again, the red dots represent the previous measurement while the blue dots show the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Writes/bob: two-node/50KiB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Slope	1.3049 ms	1.3205 ms	1.3386 ms
Throughput	36.476 MiB/s	36.977 MiB/s	37.418 MiB/s
R²	0.7328092	0.7444263	0.7287392
Mean	1.2970 ms	1.3096 ms	1.3220 ms
Std. Dev.	45.426 Âµs	63.626 Âµs	79.571 Âµs
Median	1.3110 ms	1.3194 ms	1.3240 ms
MAD	20.803 Âµs	26.170 Âµs	38.116 Âµs

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD
- Slope

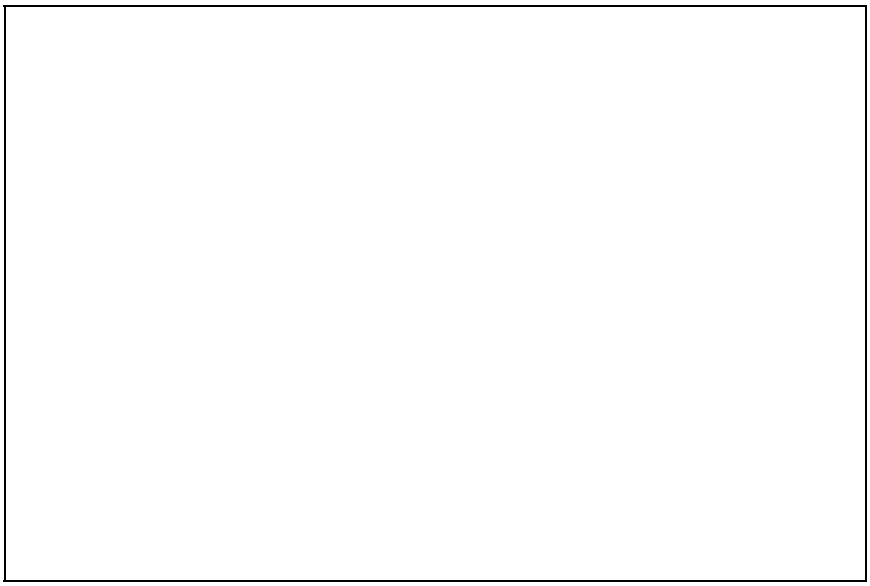
Understanding this report:

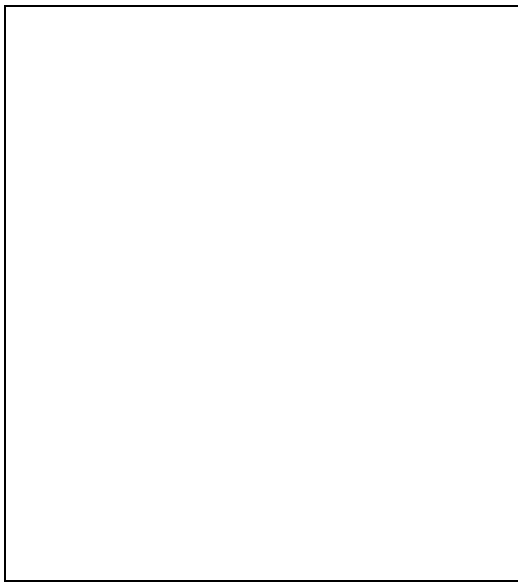
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the linear regression calculated from the measurements. Each point represents a sample, though here it shows the total time for the sample rather than time per iteration. The line is the line of best fit for these measurements.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	-1.9384%	-0.7552%	+0.4274%	(p = 0.22 > 0.05)
Change in throughput	+1.9767%	+0.7609%	-0.4256%	

No change in performance detected.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

The plot on the right shows the two regressions. Again, the red line represents the previous measurement while the blue line shows the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Writes/bob: two-node

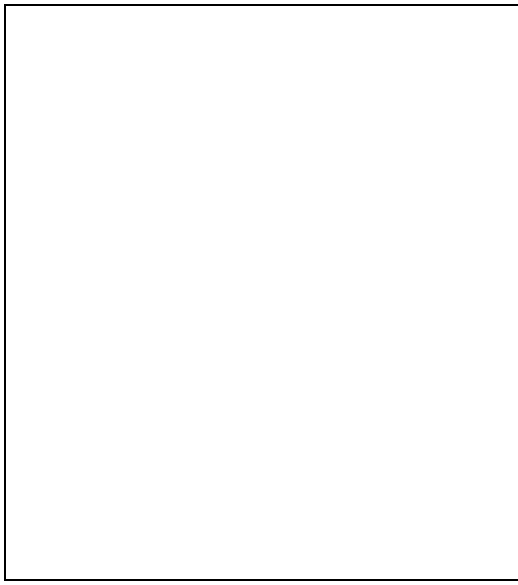
Violin Plot

This chart shows the relationship between function/parameter and iteration time. The thickness of the shaded region indicates the probability that a measurement of the given function/parameter would take a particular length of time.

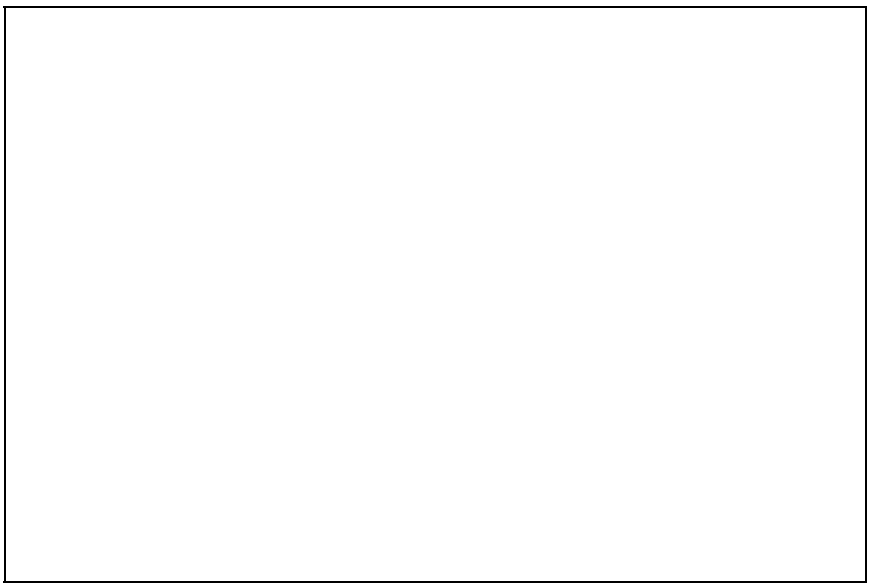
Line Chart

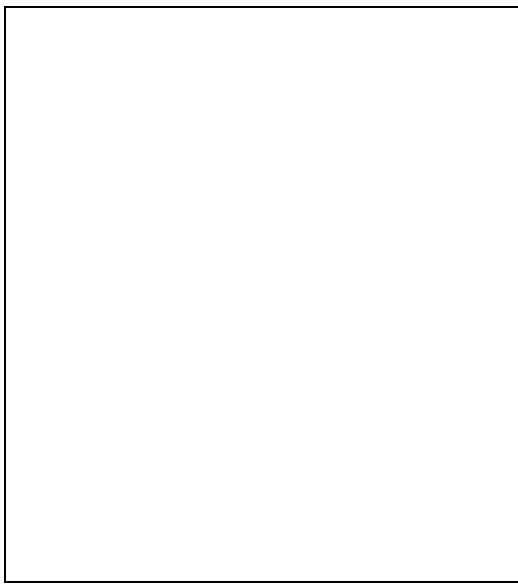
This chart shows the mean measured time for each function as the input (or the size of the input) increases.

Writes/bob: two-node/1KiB

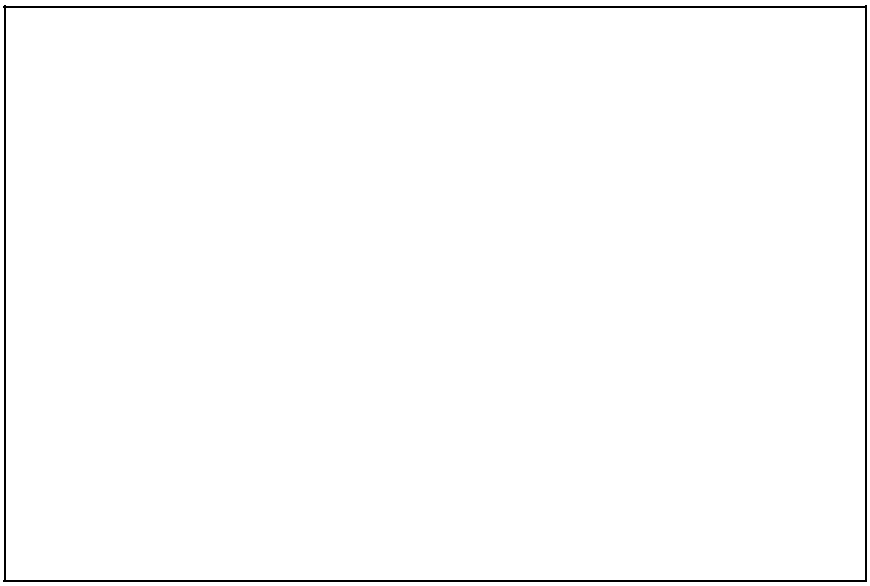


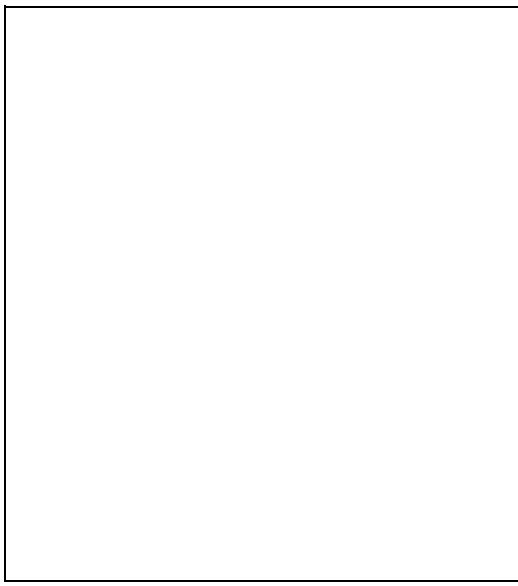
Writes/bob: two-node/50KiB



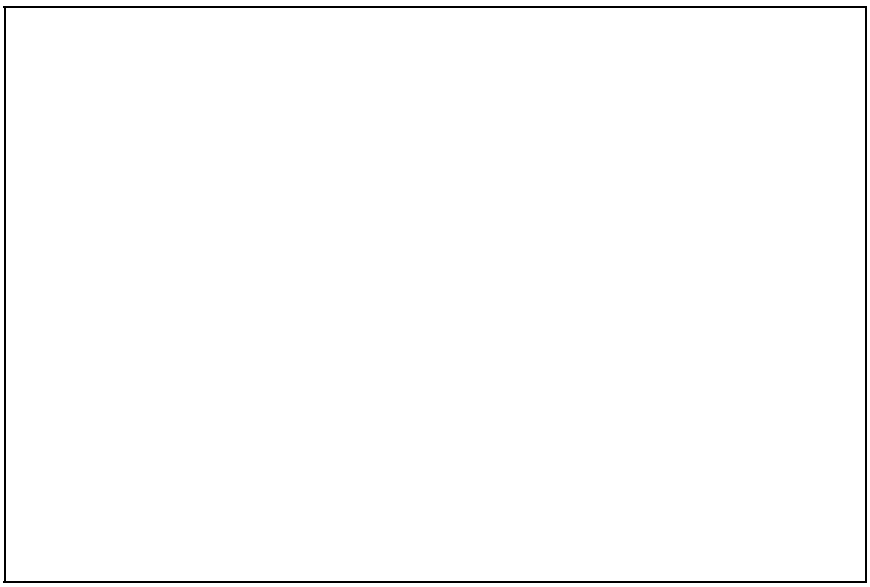


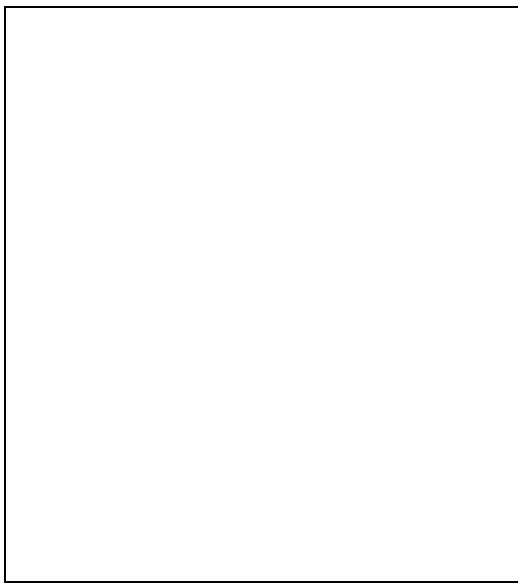
Writes/bob: two-node/1MiB





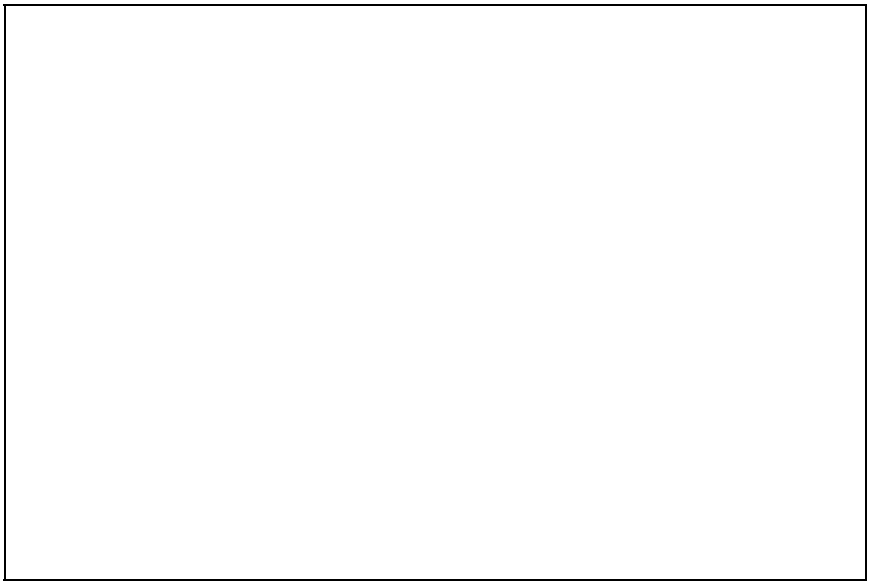
Writes/bob: two-node/100MiB

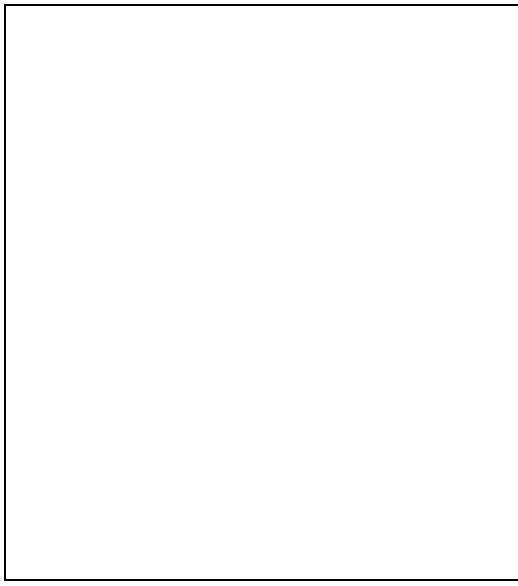




This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Writes/minio: one-node/1.05GB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Throughput	37.394 MiB/s	38.154 MiB/s	38.777 MiB/s
R^2	0.0000944	0.0000971	0.0000928
Mean	25.788 s	26.210 s	26.742 s
Std. Dev.	1.1391 s	2.4660 s	3.6412 s
Median	25.521 s	25.803 s	26.002 s
MAD	929.47 ms	1.1980 s	1.5373 s

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD

Understanding this report:

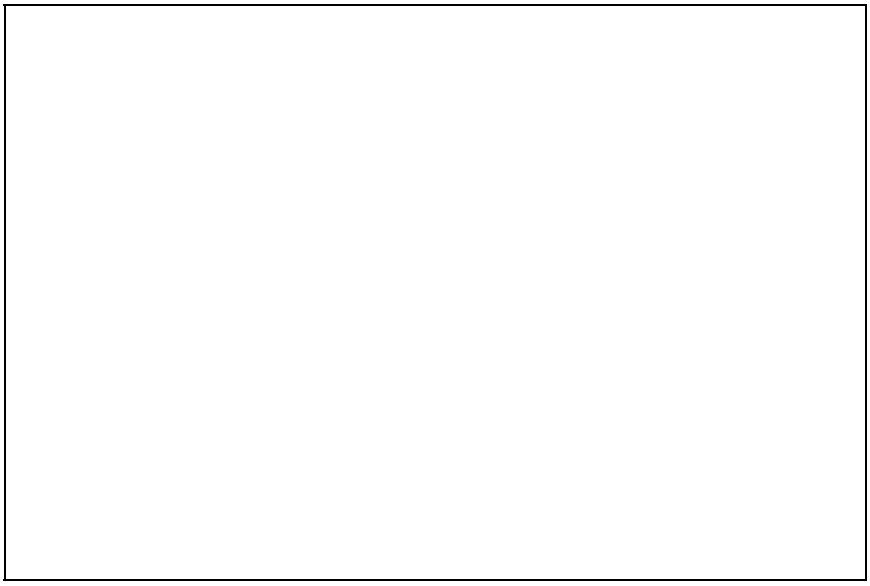
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

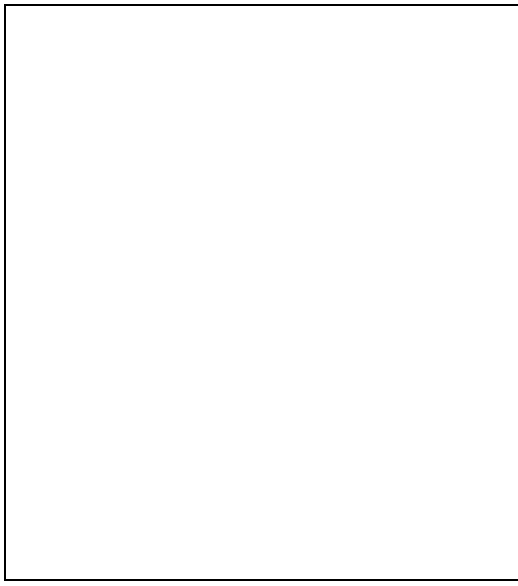
The plot on the right shows the average time per iteration for the samples. Each point represents one sample.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Writes/minio: one-node/100MiB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Throughput	31.714 MiB/s	33.142 MiB/s	34.585 MiB/s
R^2	0.0138612	0.0143456	0.0137835
Mean	2.8915 s	3.0173 s	3.1532 s
Std. Dev.	518.99 ms	671.52 ms	834.29 ms
Median	2.6937 s	2.8670 s	3.0105 s
MAD	399.67 ms	534.57 ms	700.82 ms

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD

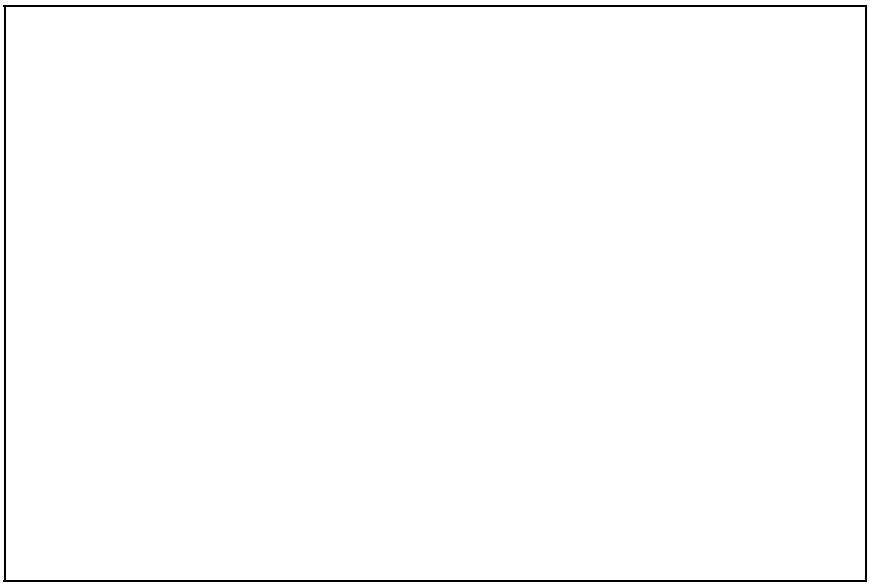
Understanding this report:

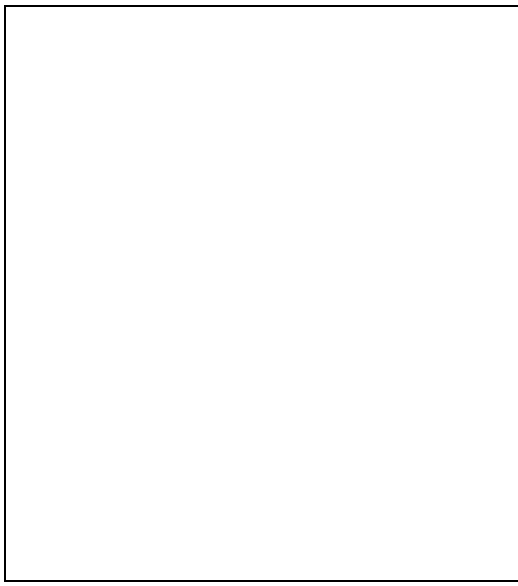
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the average time per iteration for the samples. Each point represents one sample.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	-3.9970%	+1.8268%	+7.7011%	(p = 0.53 > 0.05)
Change in throughput	+4.1635%	-1.7940%	-7.1505%	

No change in performance detected.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

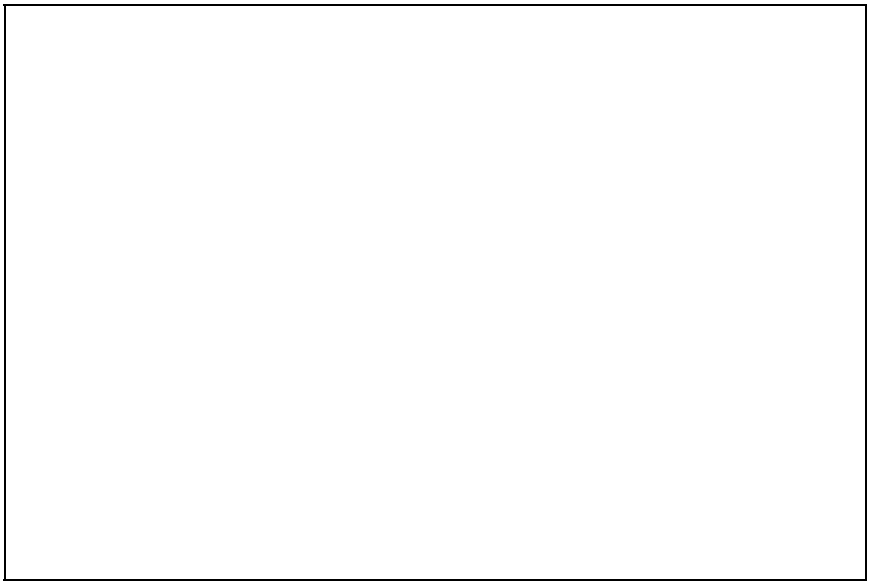
The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

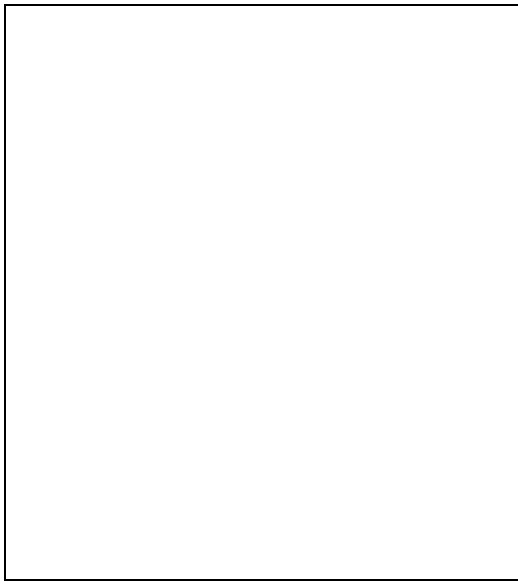
The plot on the right shows the iteration times for the two measurements. Again, the red dots represent the previous measurement while the blue dots show the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Writes/minio: one-node/1KiB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Throughput	263.56 KiB/s	324.26 KiB/s	404.98 KiB/s
R²	0.0012282	0.0012681	0.0012154
Mean	2.4692 ms	3.0839 ms	3.7943 ms
Std. Dev.	2.1836 ms	3.4273 ms	4.4545 ms
Median	1.8385 ms	1.8554 ms	1.8768 ms
MAD	95.286 Âµs	126.11 Âµs	156.32 Âµs

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD

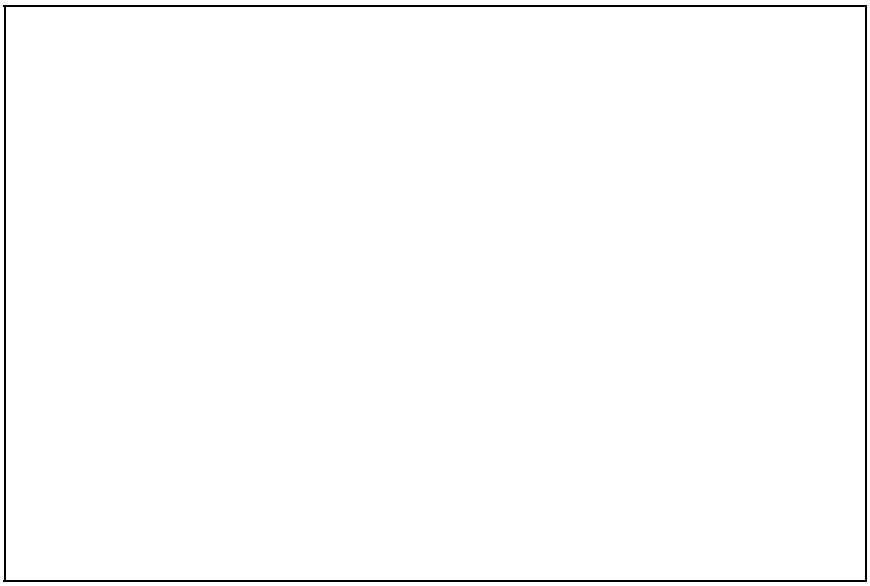
Understanding this report:

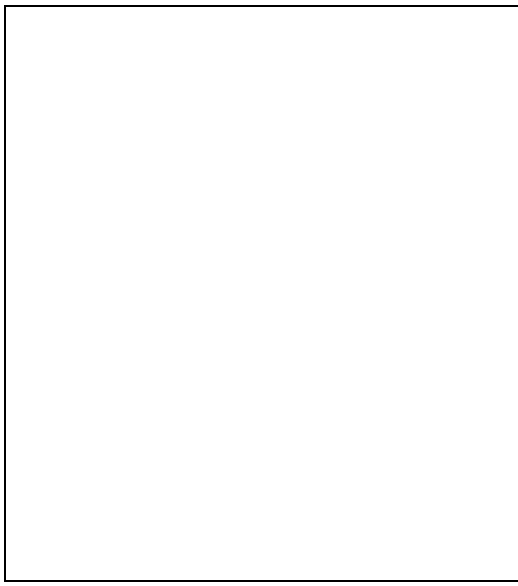
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the average time per iteration for the samples. Each point represents one sample.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	-16.392%	+14.995%	+57.410%	(p = 0.40 > 0.05)
Change in throughput	+19.606%	-13.040%	-36.472%	

No change in performance detected.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

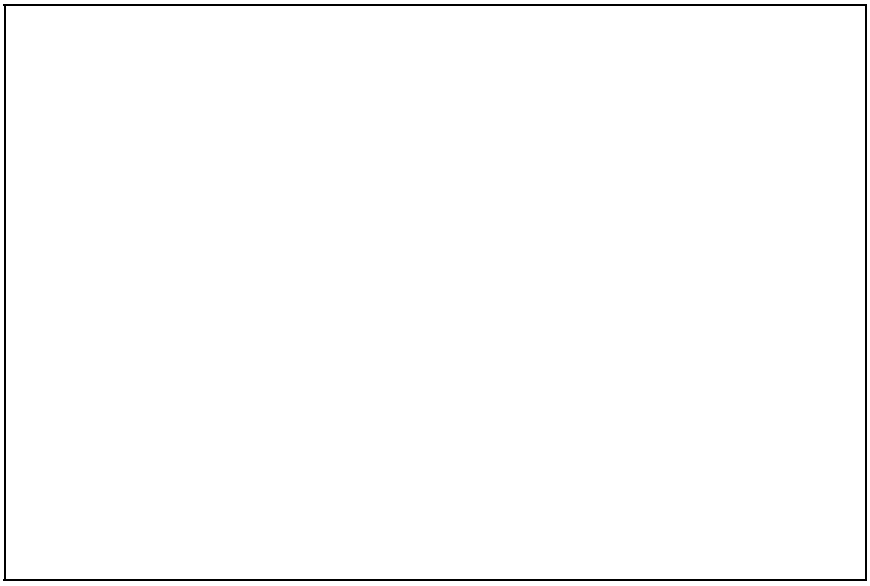
The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

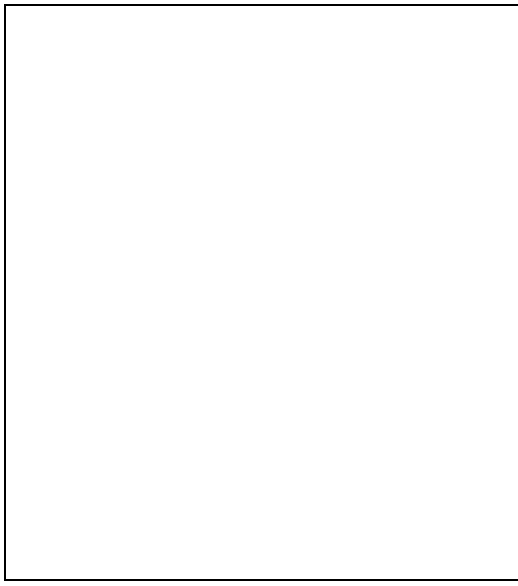
The plot on the right shows the iteration times for the two measurements. Again, the red dots represent the previous measurement while the blue dots show the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Writes/minio: one-node/1MiB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Throughput	14.306 MiB/s	19.845 MiB/s	27.307 MiB/s
R²	0.0009776	0.0010025	0.0009537
Mean	36.620 ms	50.390 ms	69.898 ms
Std. Dev.	33.833 ms	86.671 ms	137.20 ms
Median	22.656 ms	22.776 ms	23.560 ms
MAD	366.30 Âµs	642.65 Âµs	1.8153 ms

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD

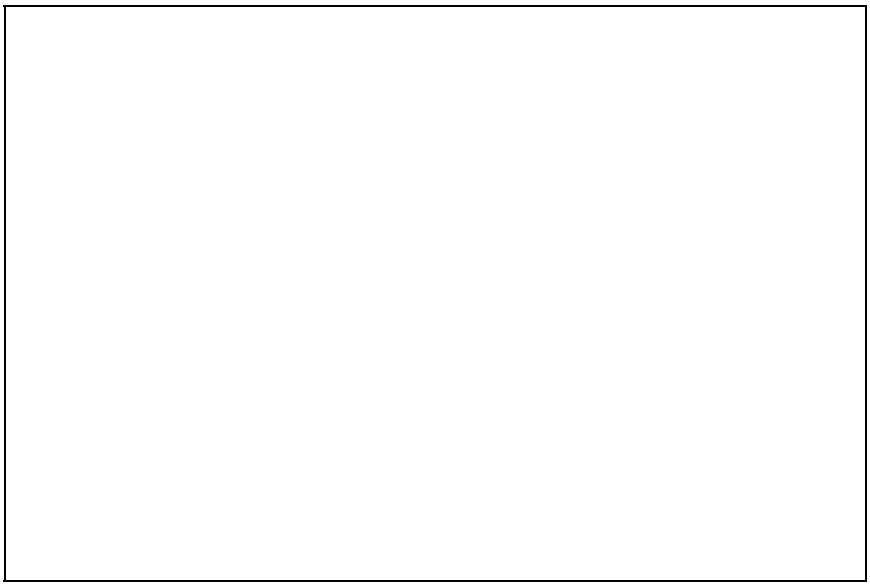
Understanding this report:

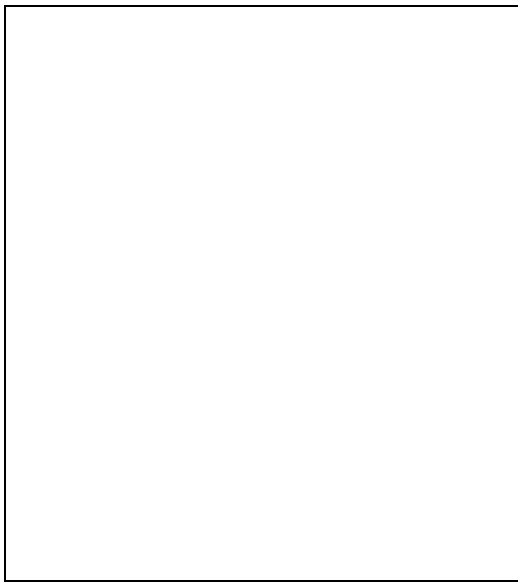
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the average time per iteration for the samples. Each point represents one sample.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	-43.664%	-18.153%	+19.169%	(p = 0.32 > 0.05)
Change in throughput	+77.507%	+22.179%	-16.086%	

No change in performance detected.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

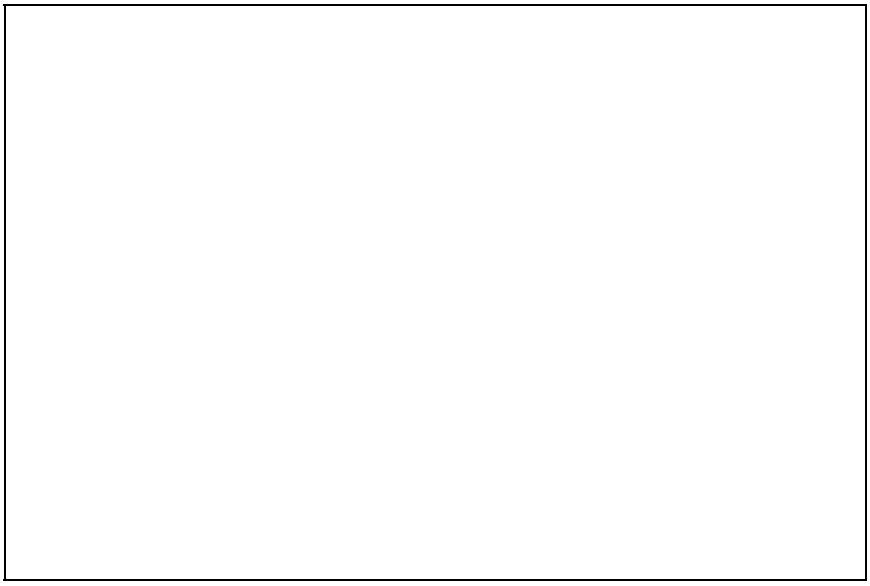
The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

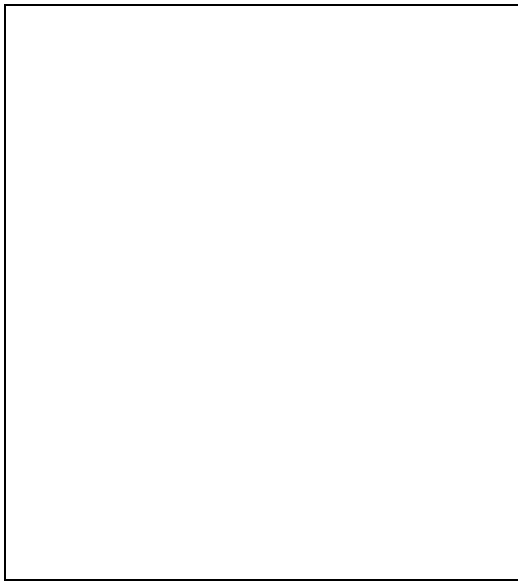
The plot on the right shows the iteration times for the two measurements. Again, the red dots represent the previous measurement while the blue dots show the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Writes/minio: one-node/50KiB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Throughput	6.2353 MiB/s	8.4048 MiB/s	11.550 MiB/s
R²	0.0004500	0.0004630	0.0004422
Mean	4.2276 ms	5.8096 ms	7.8310 ms
Std. Dev.	3.3542 ms	9.3682 ms	13.465 ms
Median	3.6191 ms	3.6685 ms	3.7202 ms
MAD	106.69 Âµs	148.72 Âµs	192.96 Âµs

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD

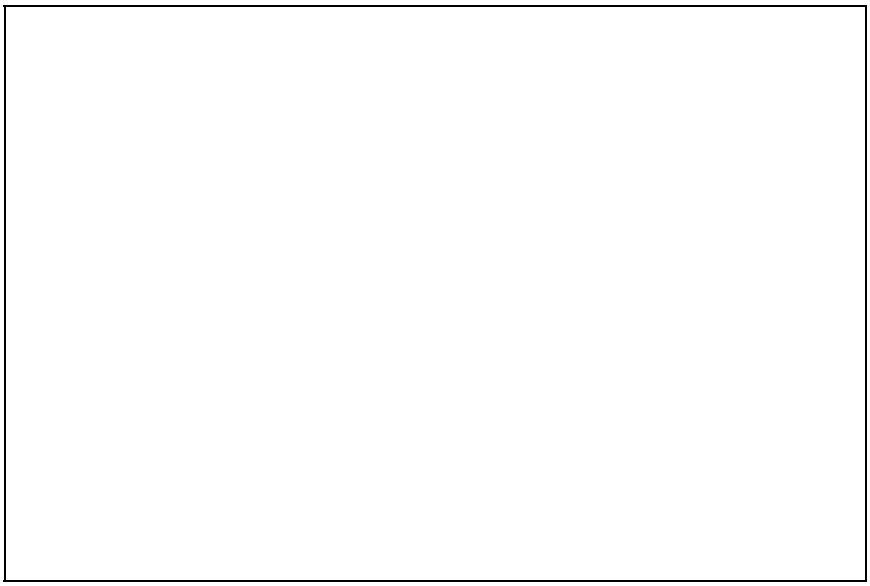
Understanding this report:

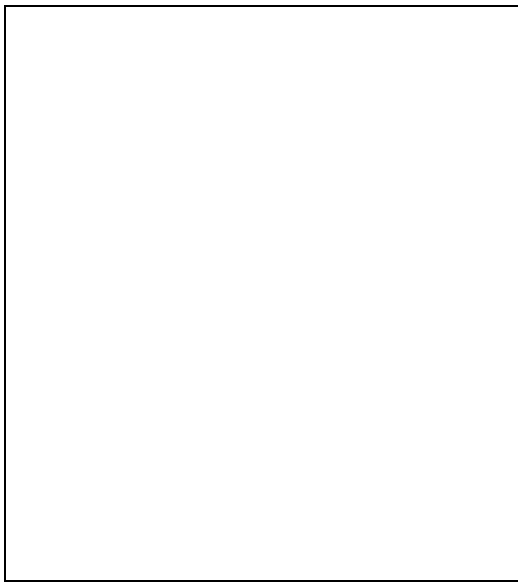
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the average time per iteration for the samples. Each point represents one sample.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	-36.587%	-2.2761%	+53.654%	(p = 0.93 > 0.05)
Change in throughput	+57.696%	+2.3291%	-34.919%	

No change in performance detected.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

The plot on the right shows the iteration times for the two measurements. Again, the red dots represent the previous measurement while the blue dots show the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Writes/minio: one-node

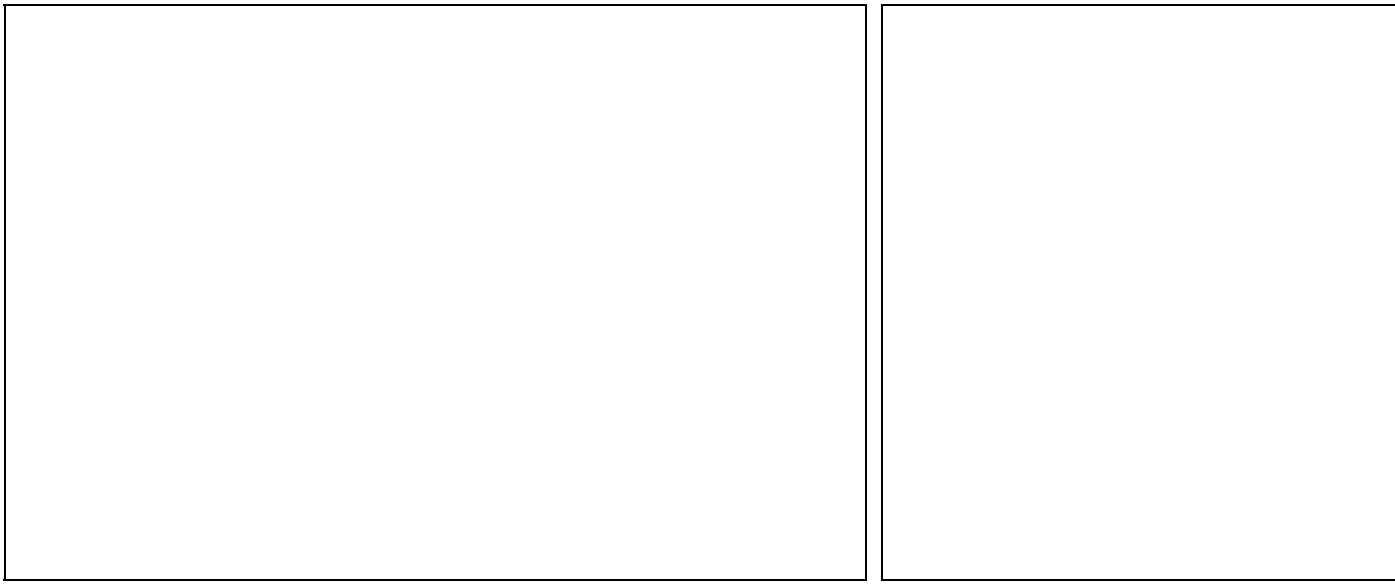
Violin Plot

This chart shows the relationship between function/parameter and iteration time. The thickness of the shaded region indicates the probability that a measurement of the given function/parameter would take a particular length of time.

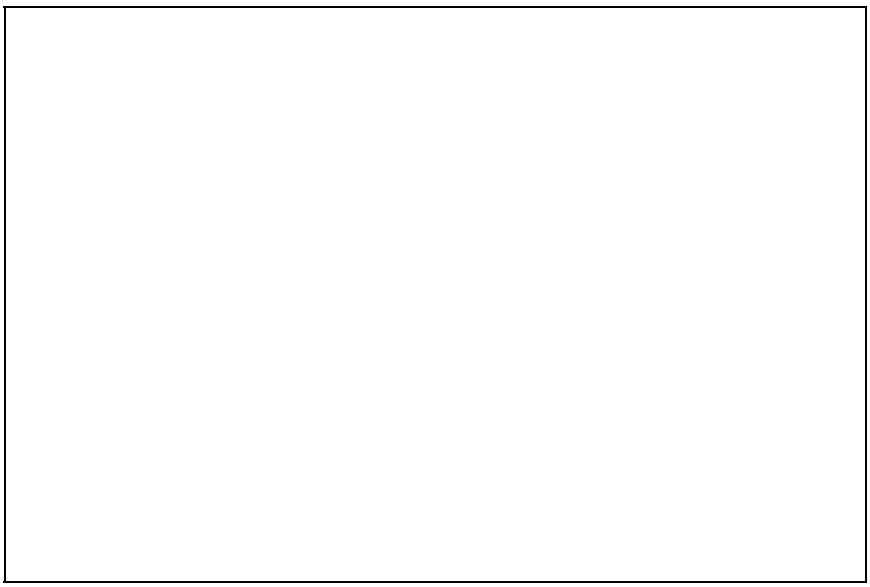
Line Chart

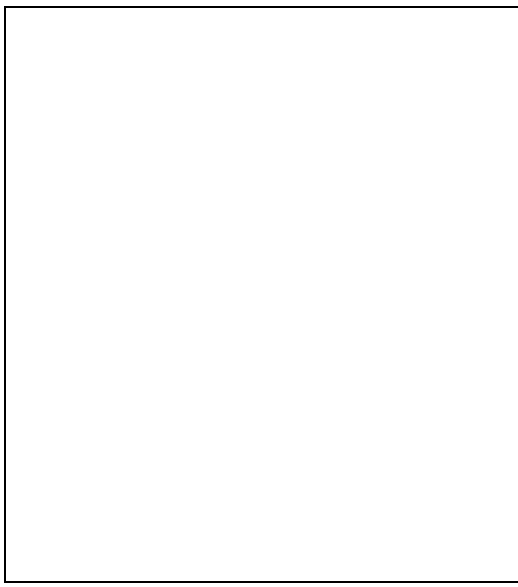
This chart shows the mean measured time for each function as the input (or the size of the input) increases.

Writes/minio: one-node/1KiB

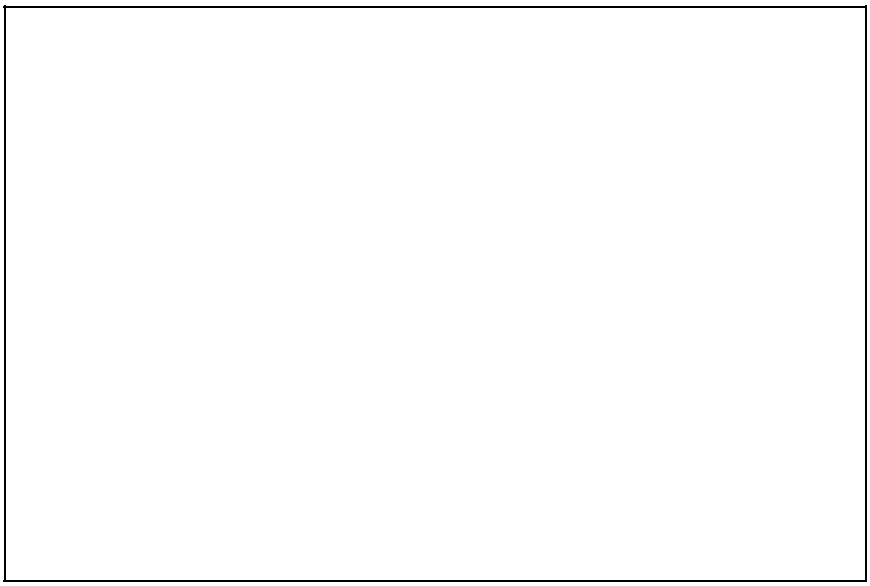


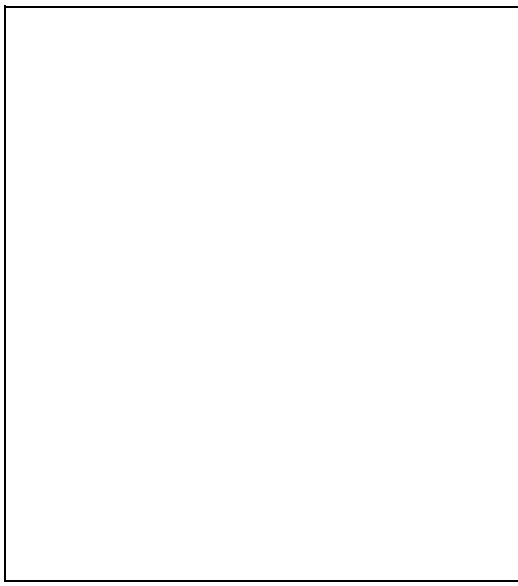
Writes/minio: one-node/50KiB



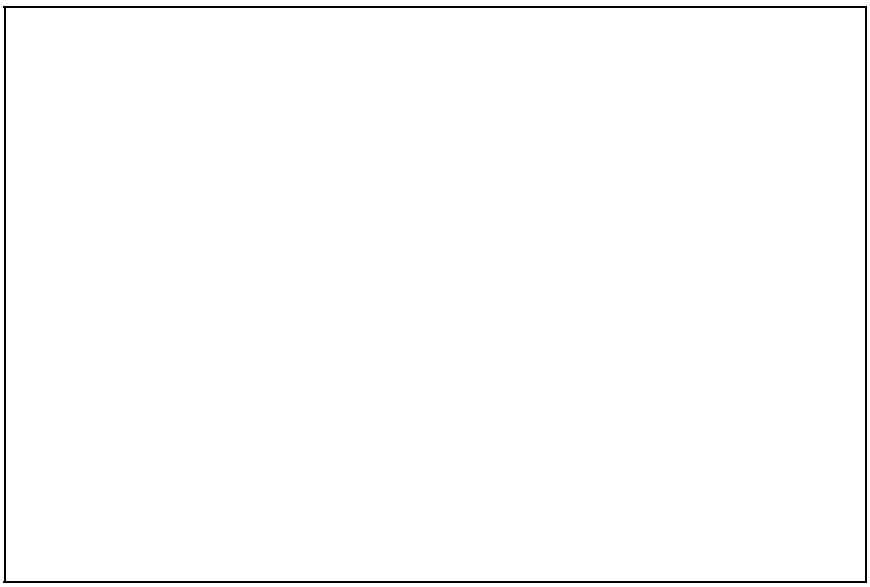


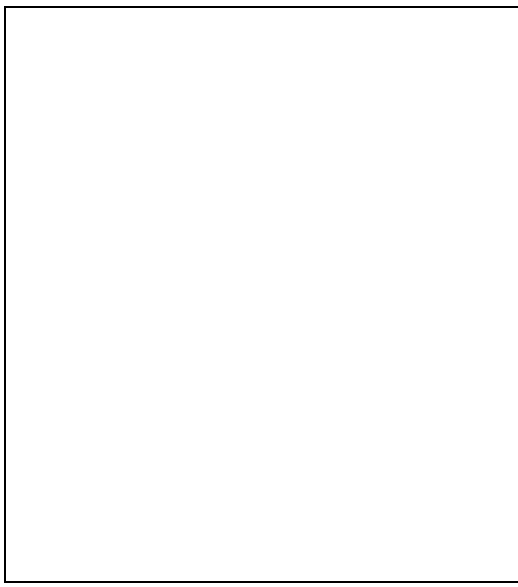
Writes/minio: one-node/1MiB





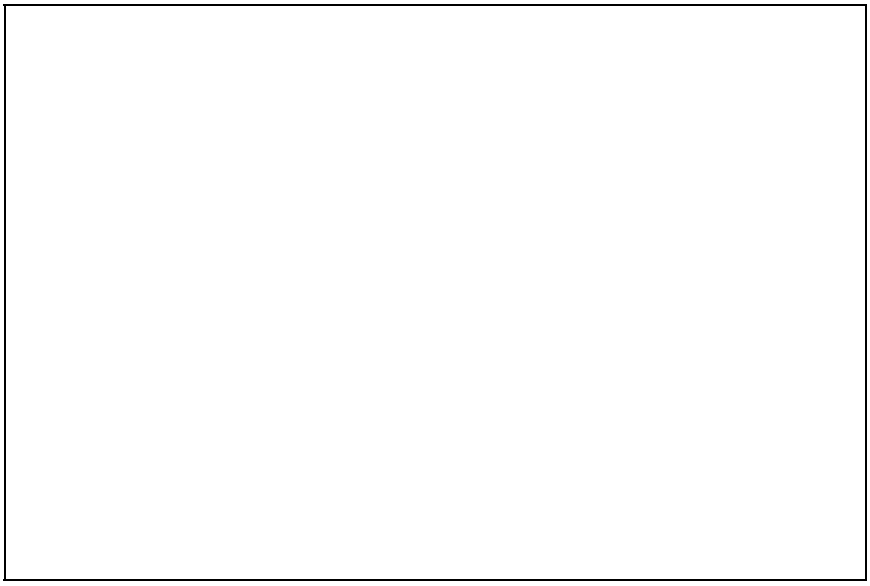
Writes/minio: one-node/100MiB

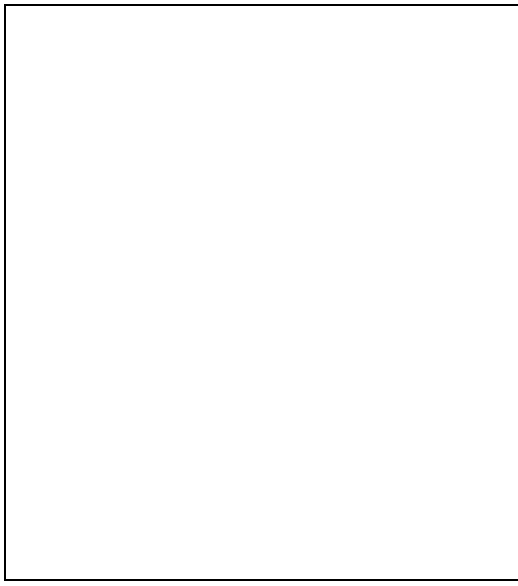




This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Writes/minio: two-node/1.05GB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Throughput	33.157 MiB/s	33.428 MiB/s	33.692 MiB/s
R^2	0.0012799	0.0013272	0.0012761
Mean	29.680 s	29.915 s	30.159 s
Std. Dev.	1.0199 s	1.2260 s	1.3966 s
Median	29.387 s	29.494 s	29.685 s
MAD	599.78 ms	815.76 ms	1.0260 s

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD

Understanding this report:

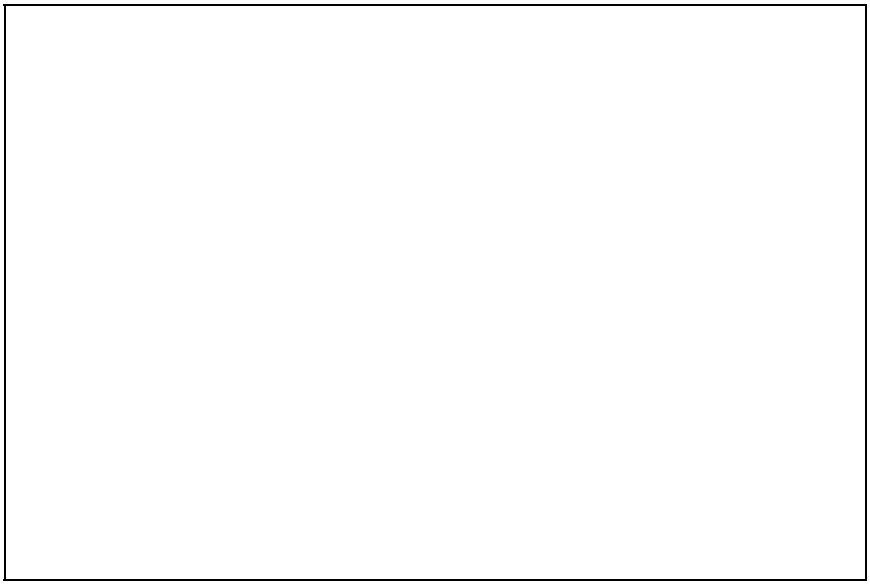
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

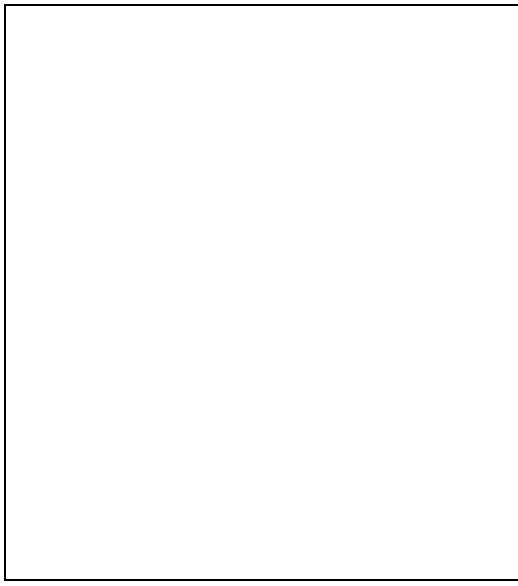
The plot on the right shows the average time per iteration for the samples. Each point represents one sample.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Writes/minio: two-node/100MiB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Throughput	26.911 MiB/s	28.071 MiB/s	29.260 MiB/s
R^2	0.0028623	0.0029653	0.0028501
Mean	3.4177 s	3.5625 s	3.7159 s
Std. Dev.	622.09 ms	765.80 ms	895.46 ms
Median	3.1578 s	3.2403 s	3.5259 s
MAD	384.59 ms	514.38 ms	802.21 ms

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD

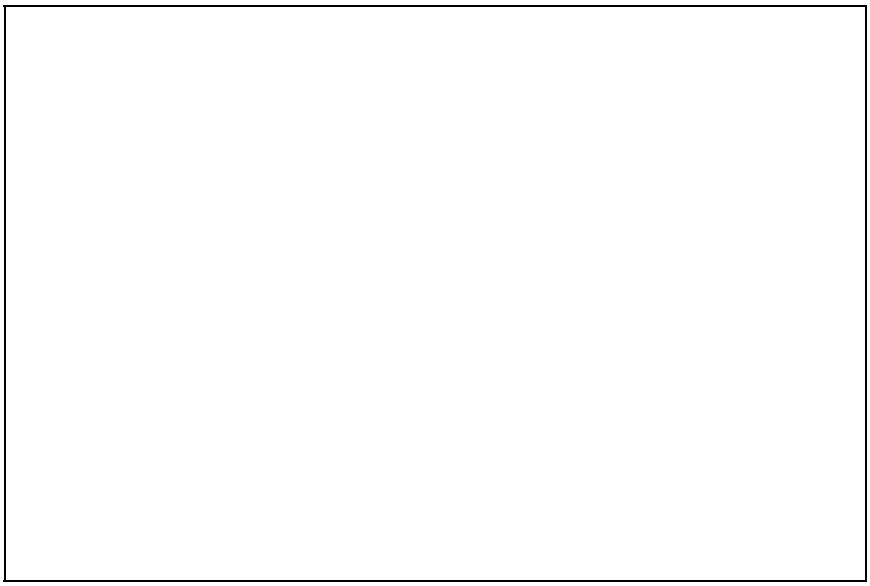
Understanding this report:

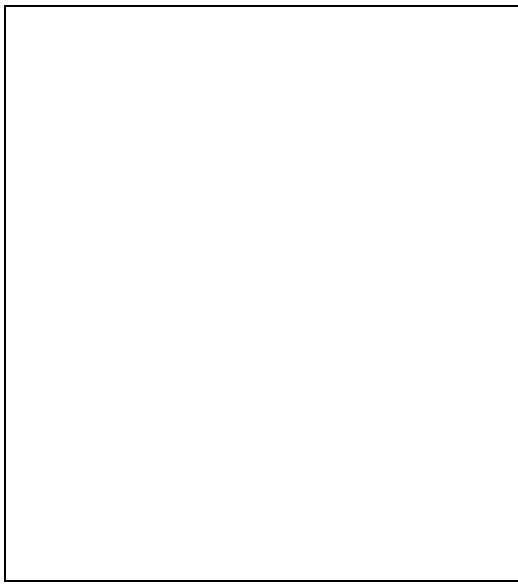
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the average time per iteration for the samples. Each point represents one sample.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	-15.488%	-9.7390%	-3.7418%	(p = 0.00 < 0.05)
Change in throughput	+18.326%	+10.790%	+3.8873%	

Performance has improved.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

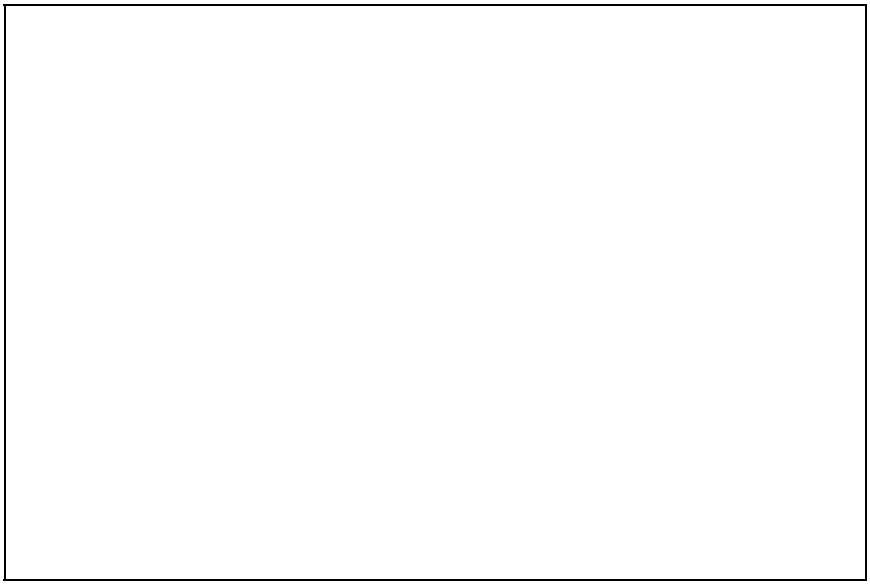
The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

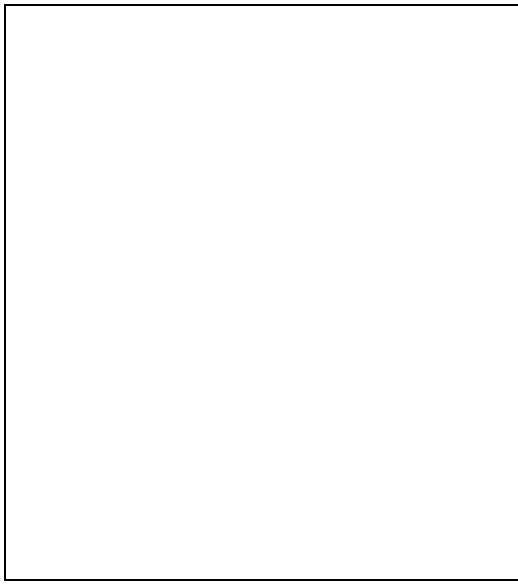
The plot on the right shows the iteration times for the two measurements. Again, the red dots represent the previous measurement while the blue dots show the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Writes/minio: two-node/1KiB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Throughput	47.490 KiB/s	75.217 KiB/s	142.64 KiB/s
R²	0.0006104	0.0006288	0.0006011
Mean	7.0105 ms	13.295 ms	21.057 ms
Std. Dev.	17.409 ms	36.286 ms	52.262 ms
Median	3.1019 ms	3.1499 ms	3.2090 ms
MAD	120.43 Âµs	174.13 Âµs	259.86 Âµs

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD

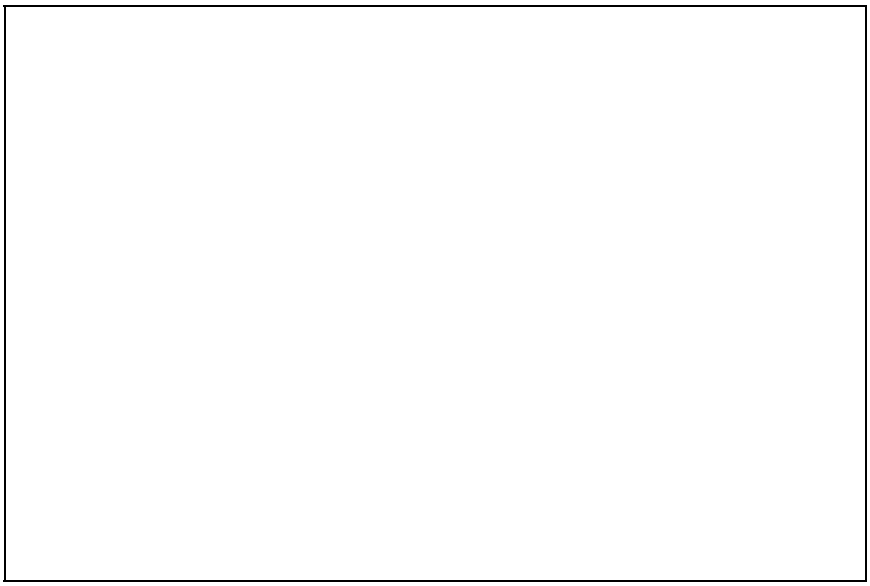
Understanding this report:

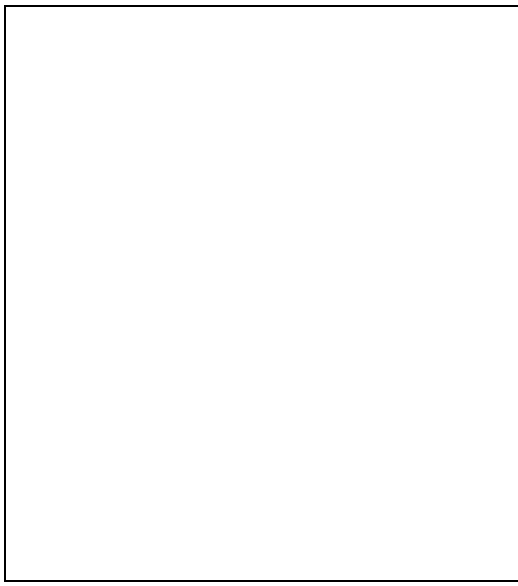
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the average time per iteration for the samples. Each point represents one sample.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	+19.787%	+152.41%	+353.67%	(p = 0.02 < 0.05)
Change in throughput	-16.518%	-60.381%	-77.958%	

Performance has regressed.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

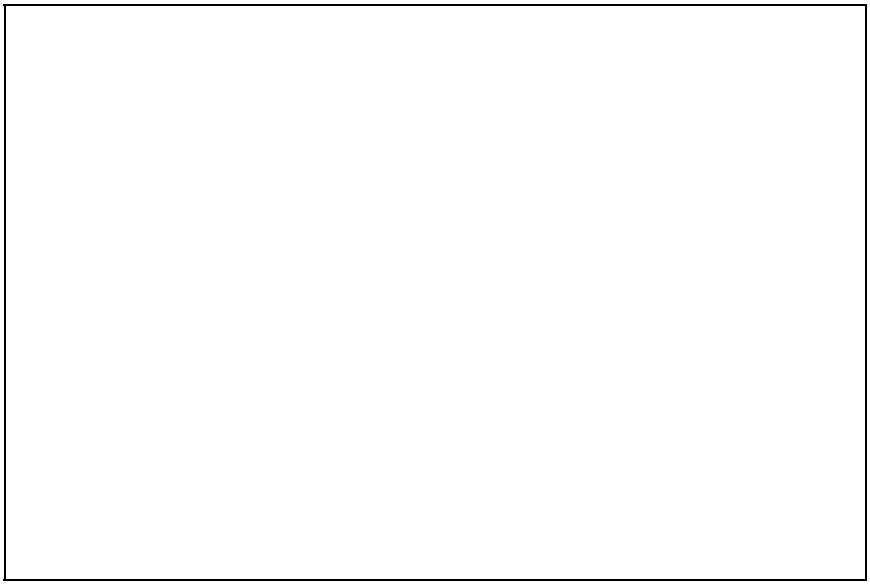
The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

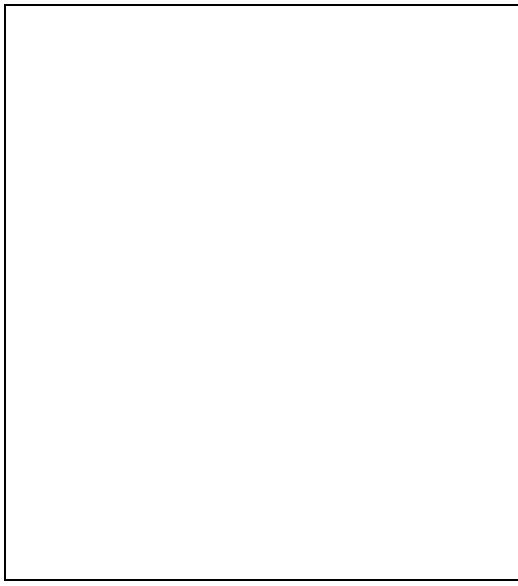
The plot on the right shows the iteration times for the two measurements. Again, the red dots represent the previous measurement while the blue dots show the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Writes/minio: two-node/1MiB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Throughput	6.4137 MiB/s	8.8151 MiB/s	12.861 MiB/s
R^2	0.0019192	0.0019797	0.0018950
Mean	77.752 ms	113.44 ms	155.92 ms
Std. Dev.	115.04 ms	201.78 ms	273.29 ms
Median	25.159 ms	25.859 ms	34.281 ms
MAD	1.4838 ms	2.6742 ms	15.462 ms

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD

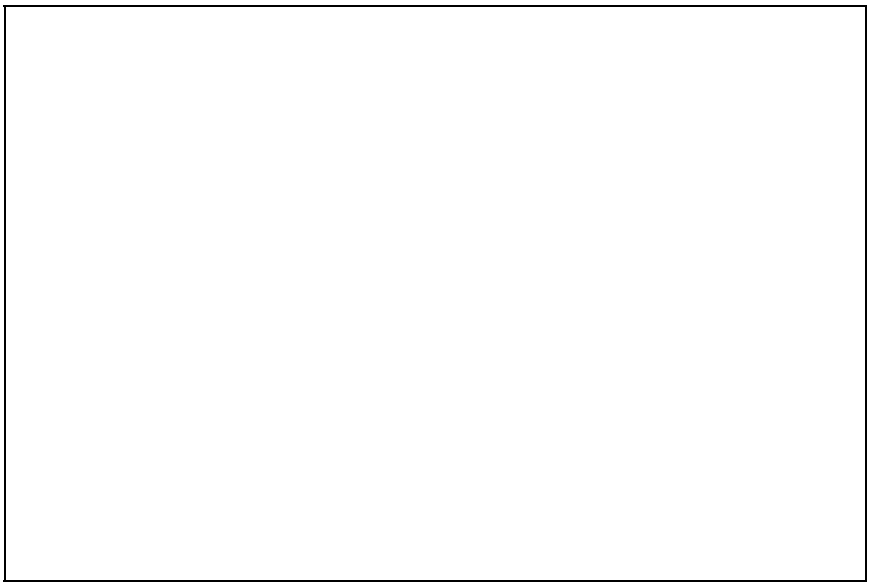
Understanding this report:

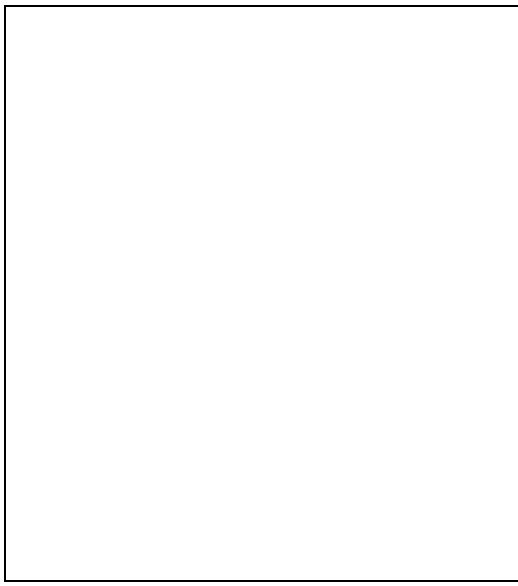
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the average time per iteration for the samples. Each point represents one sample.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	-35.213%	-1.9513%	+50.464%	(p = 0.93 > 0.05)
Change in throughput	+54.353%	+1.9902%	-33.539%	

No change in performance detected.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

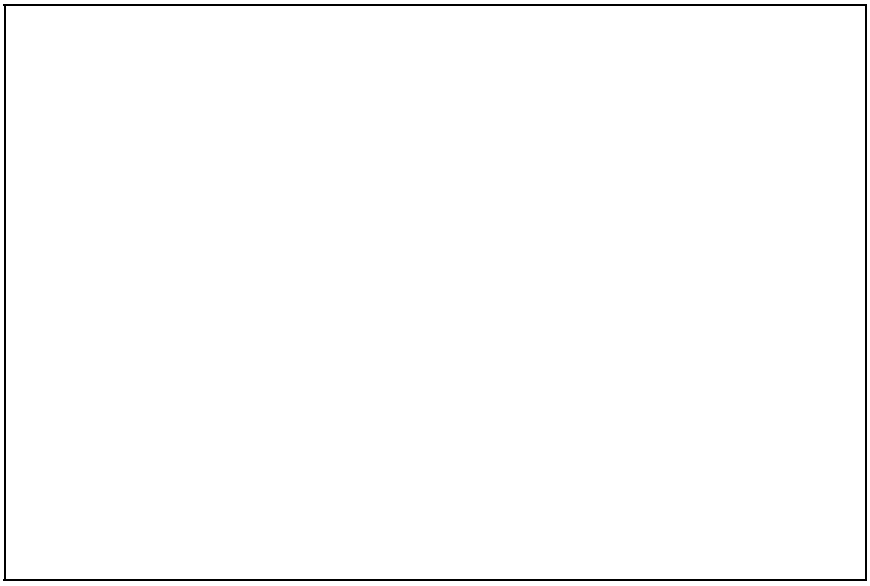
The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

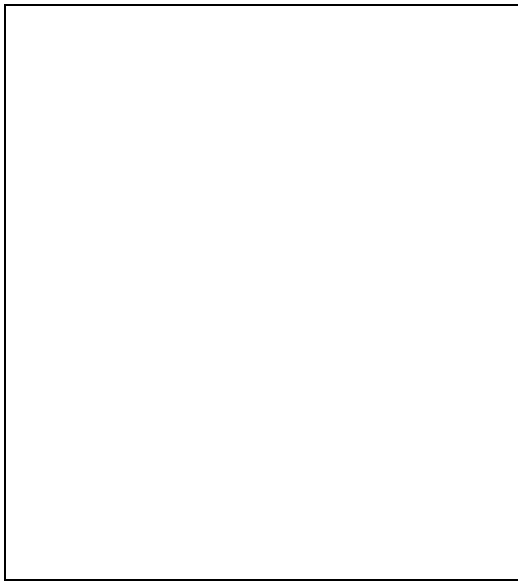
The plot on the right shows the iteration times for the two measurements. Again, the red dots represent the previous measurement while the blue dots show the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Writes/minio: two-node/50KiB





Additional Statistics:

	Lower bound	Estimate	Upper bound
Throughput	4.5926 MiB/s	6.0598 MiB/s	8.2122 MiB/s
R²	0.0007101	0.0007318	0.0007000
Mean	5.9458 ms	8.0578 ms	10.632 ms
Std. Dev.	6.3318 ms	12.134 ms	16.568 ms
Median	4.5193 ms	4.5636 ms	4.6580 ms
MAD	159.26 Âµs	209.53 Âµs	301.30 Âµs

Additional Plots:

- Typical
- Mean
- Std. Dev.
- Median
- MAD

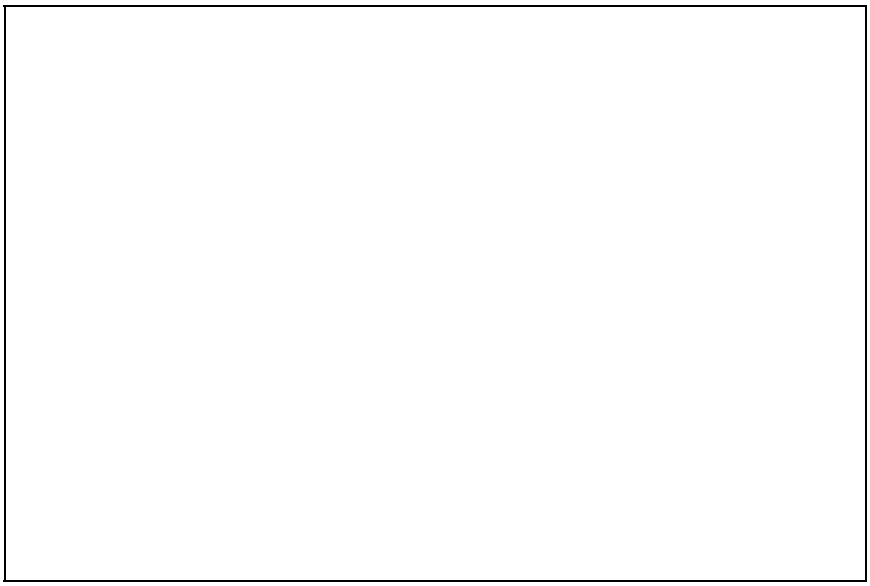
Understanding this report:

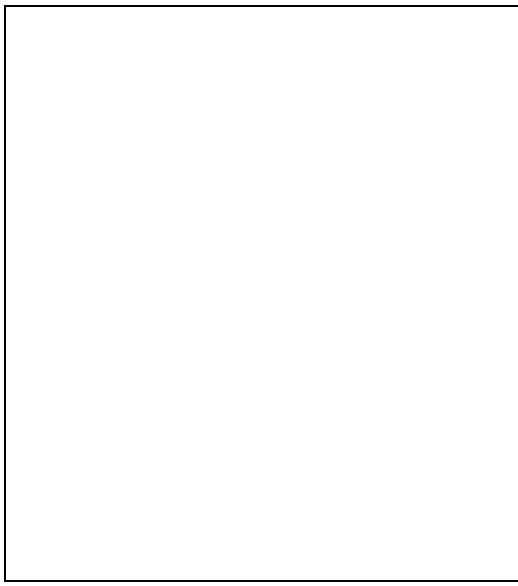
The plot on the left displays the average time per iteration for this benchmark. The shaded region shows the estimated probability of an iteration taking a certain amount of time, while the line shows the mean. Click on the plot for a larger view showing the outliers.

The plot on the right shows the average time per iteration for the samples. Each point represents one sample.

See the documentation for more details on the additional statistics.

Change Since Previous Benchmark





Additional Statistics:

	Lower bound	Estimate	Upper bound	
Change in time	-86.511%	-74.865%	-41.024%	(p = 0.01 < 0.05)
Change in throughput	+641.37%	+297.85%	+69.560%	

Performance has improved.

Additional Plots:

- Change in mean
- Change in median
- T-Test

Understanding this report:

The plot on the left shows the probability of the function taking a certain amount of time. The red curve represents the saved measurements from the last time this benchmark was run, while the blue curve shows the measurements from this run. The lines represent the mean time per iteration. Click on the plot for a larger view.

The plot on the right shows the iteration times for the two measurements. Again, the red dots represent the previous measurement while the blue dots show the current measurement.

See the documentation for more details on the additional statistics.

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Writes/minio: two-node

Violin Plot

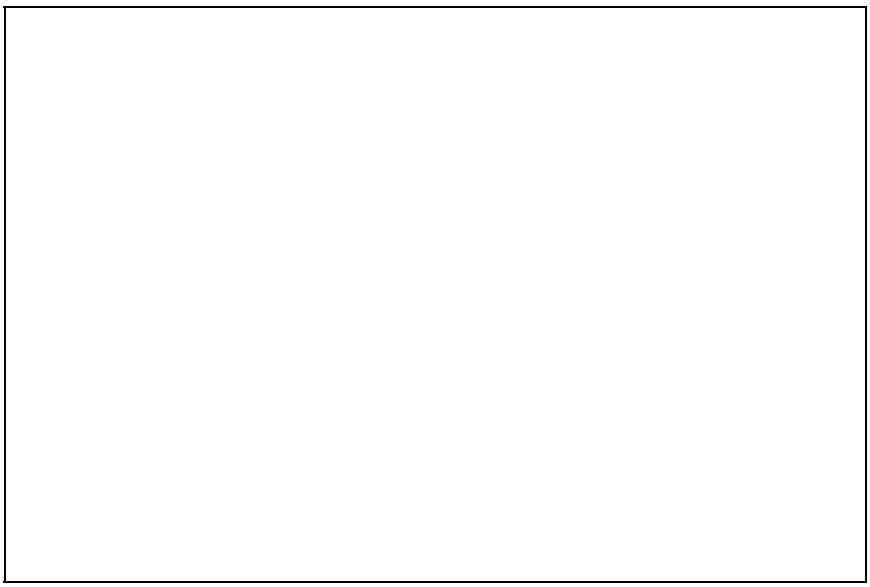
This chart shows the relationship between function/parameter and iteration time. The thickness of the shaded region indicates the probability that a measurement of the given function/parameter would take a particular length of time.

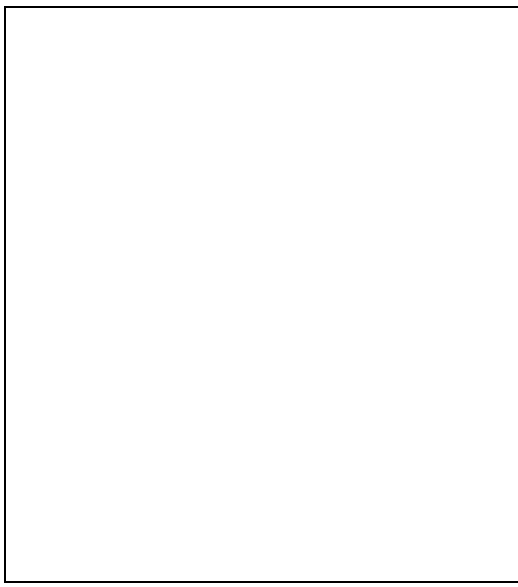
Line Chart

This chart shows the mean measured time for each function as the input (or the size of the input) increases.

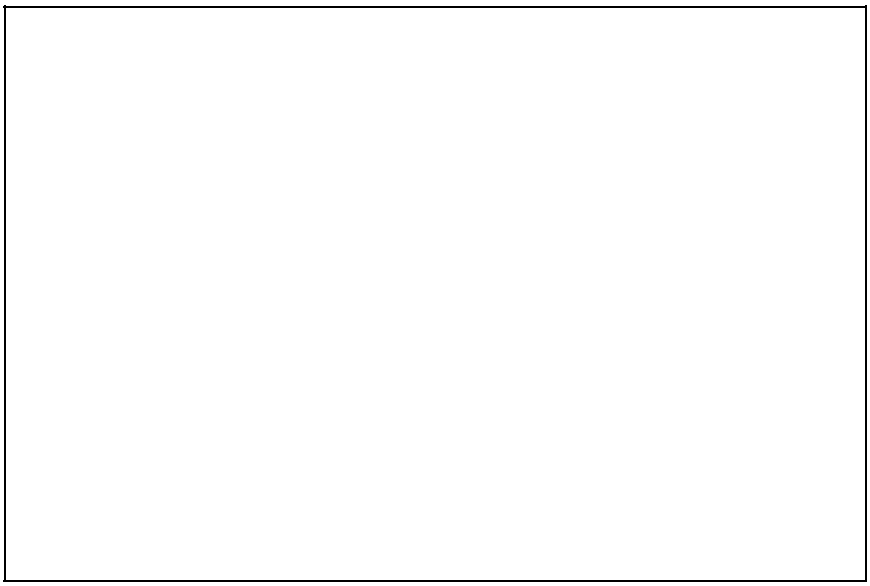
Writes/minio: two-node/1KiB

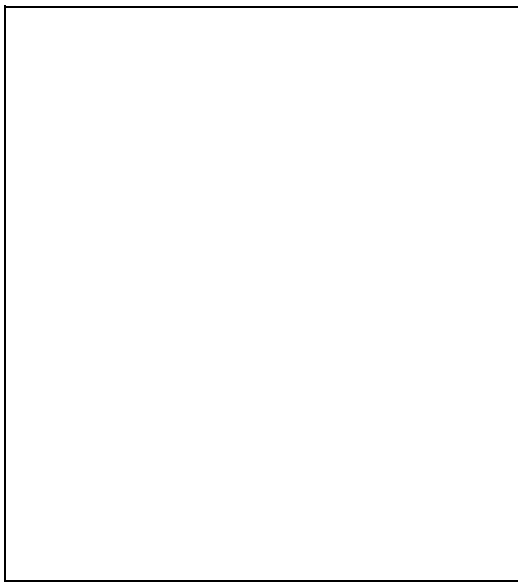
Writes/minio: two-node/50KiB



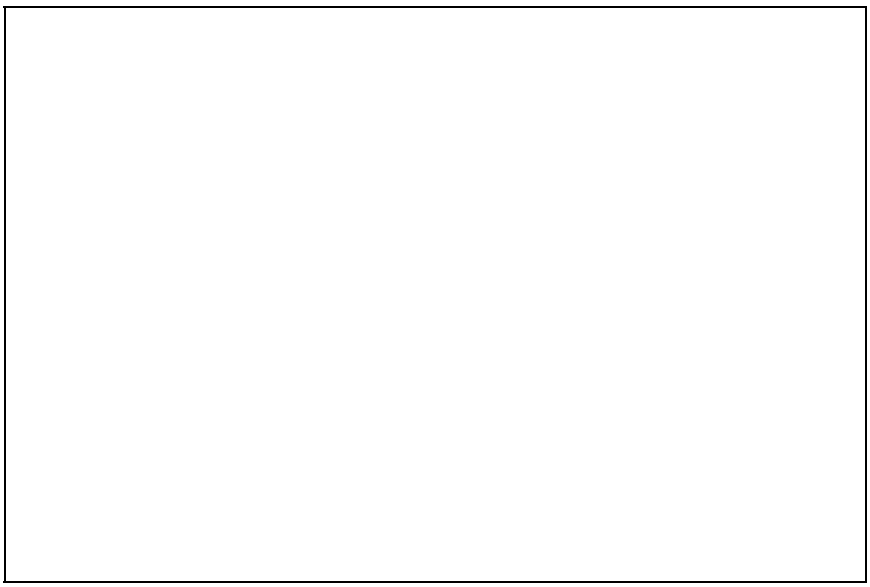


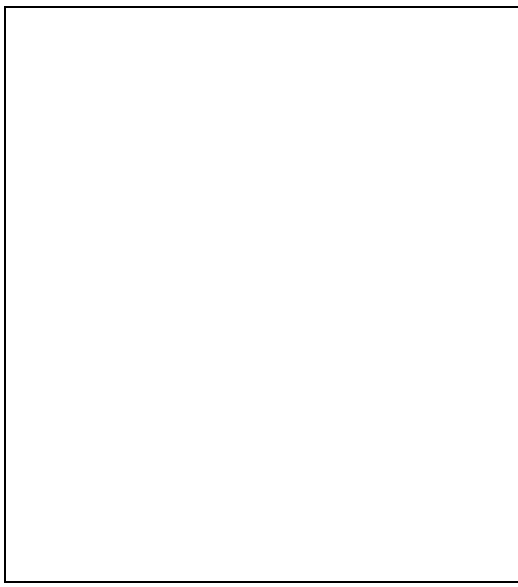
Writes/minio: two-node/1MiB





Writes/minio: two-node/100MiB





This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Writes

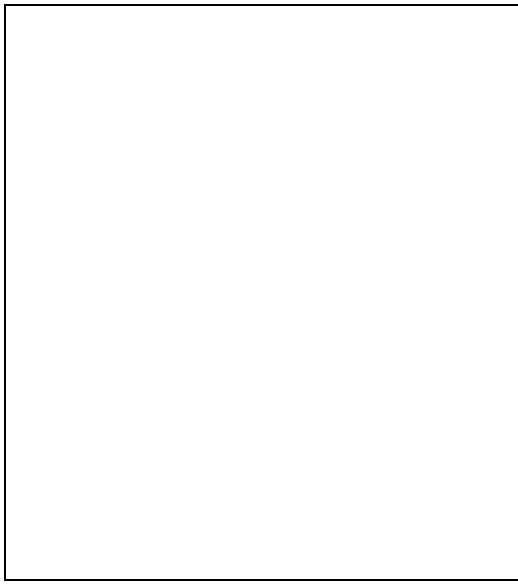
Violin Plot

This chart shows the relationship between function/parameter and iteration time. The thickness of the shaded region indicates the probability that a measurement of the given function/parameter would take a particular length of time.

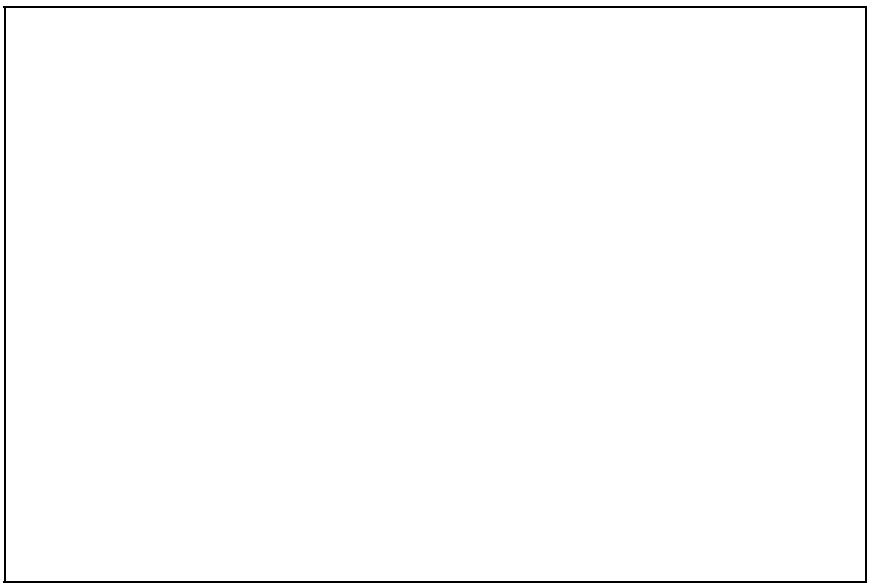
Line Chart

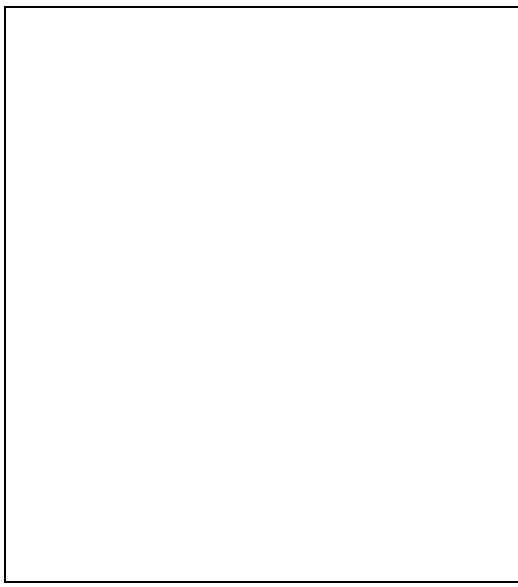
This chart shows the mean measured time for each function as the input (or the size of the input) increases.

Writes/bob: one-node/100MiB

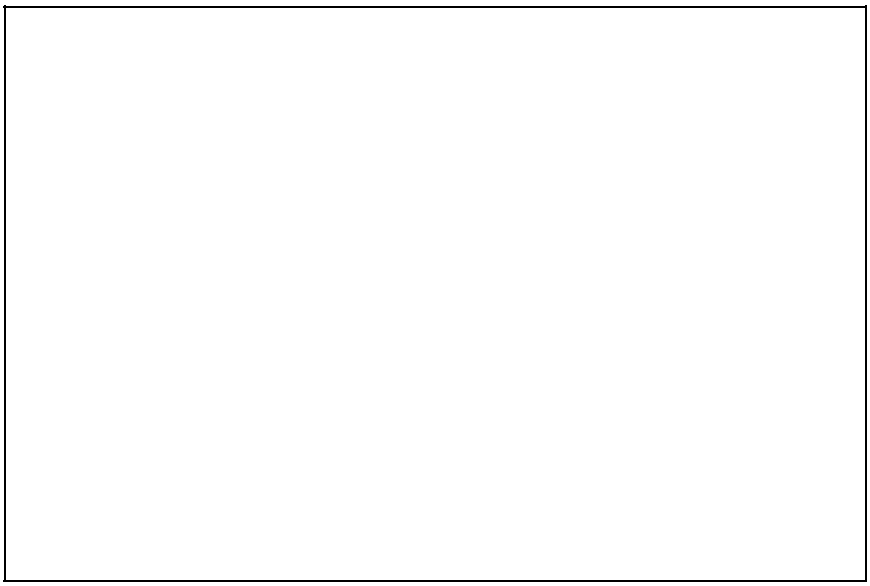


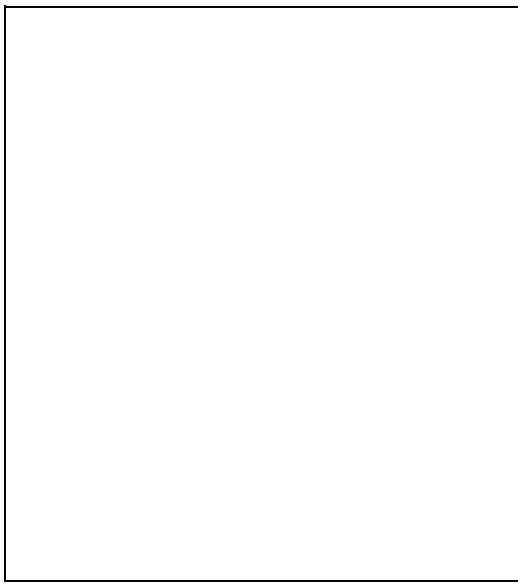
Writes/bob: one-node/1KiB



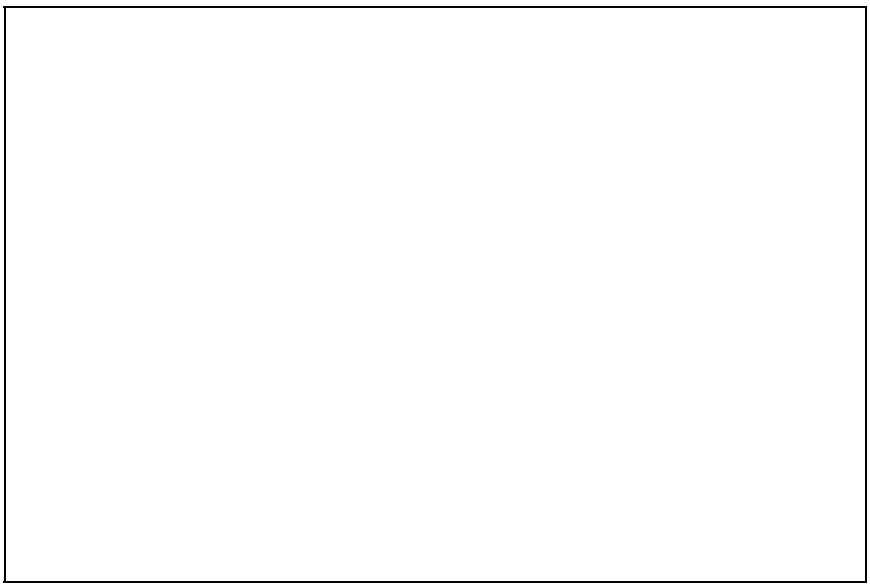


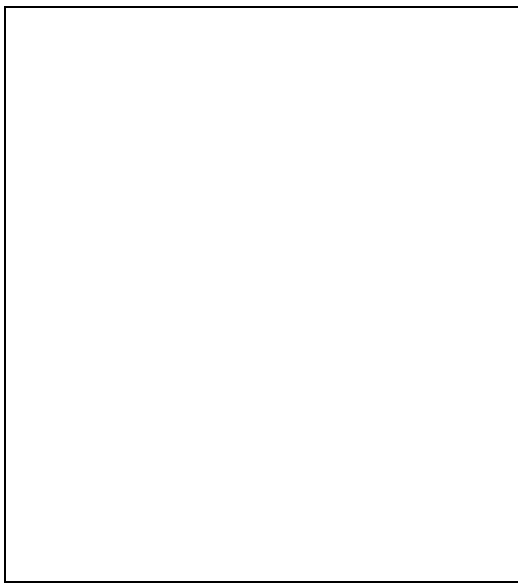
Writes/bob: one-node/1MiB



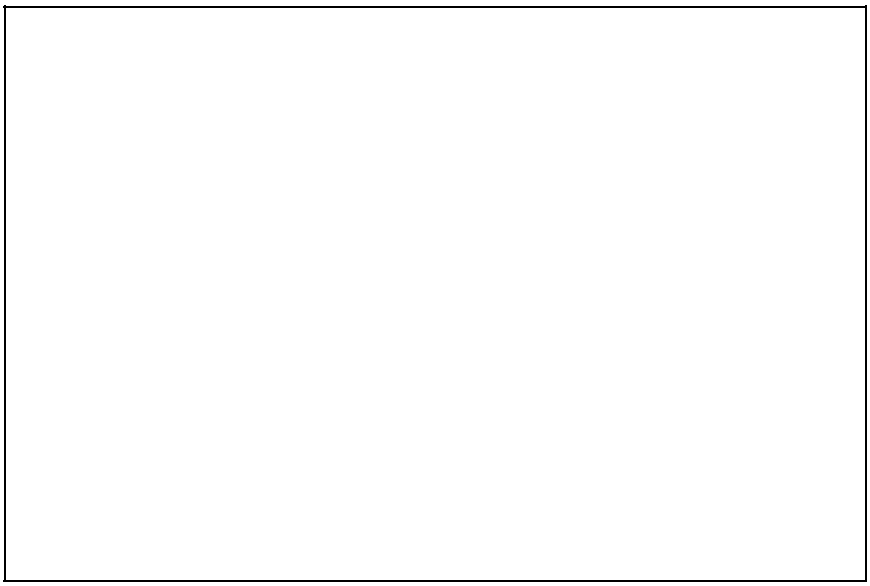


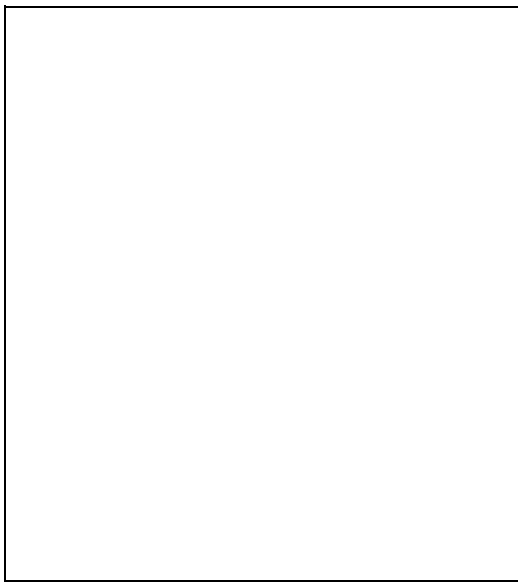
Writes/bob: one-node/50KiB



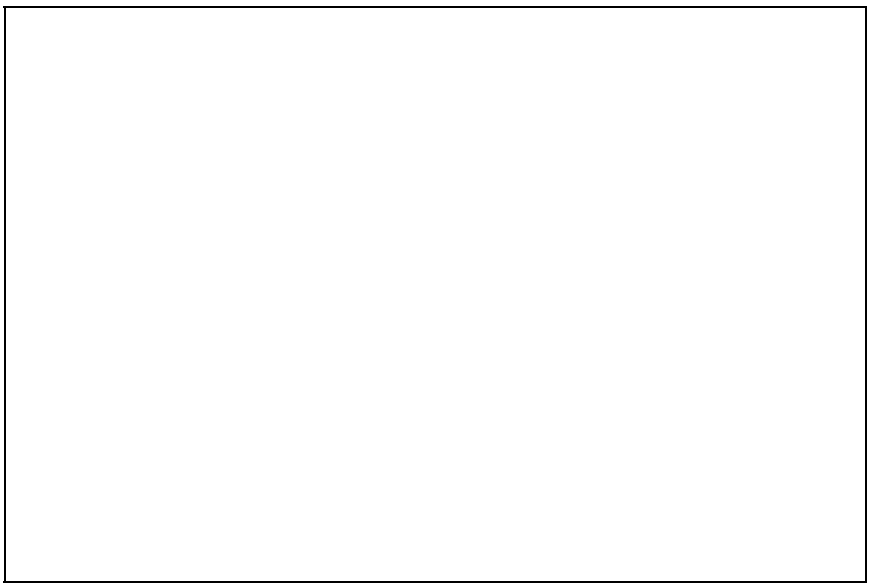


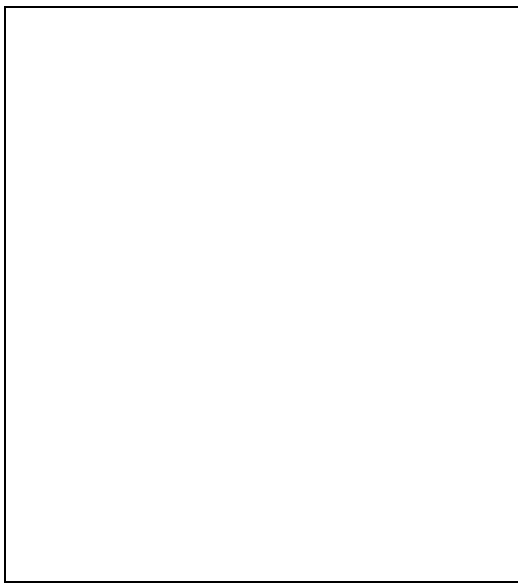
Writes/bob: two-node/100MiB



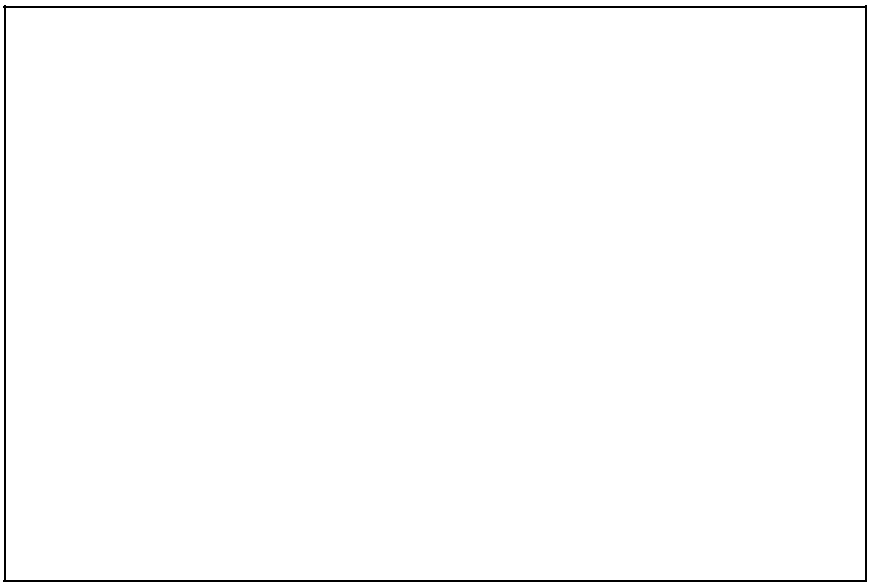


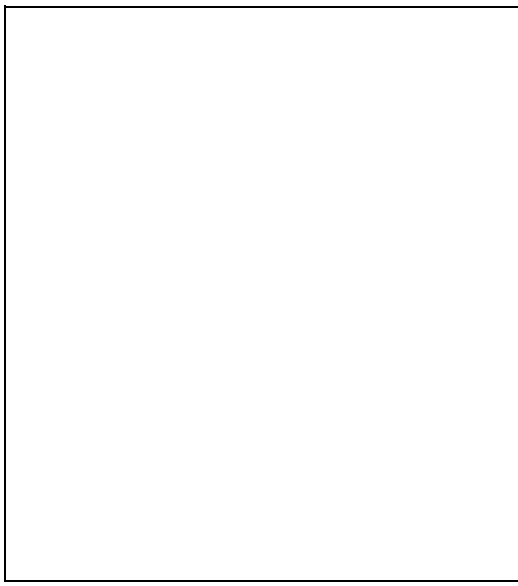
Writes/bob: two-node/1KiB



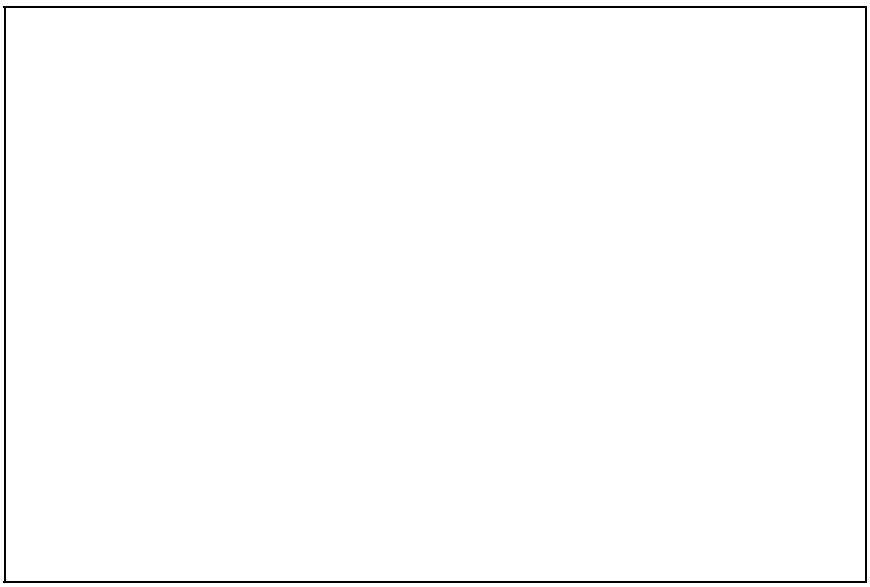


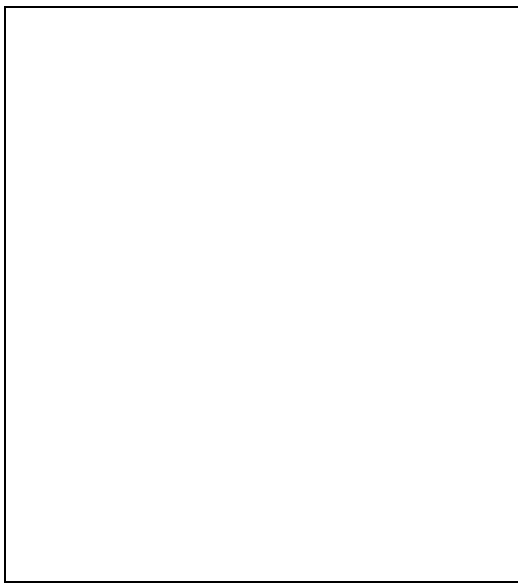
Writes/bob: two-node/1MiB



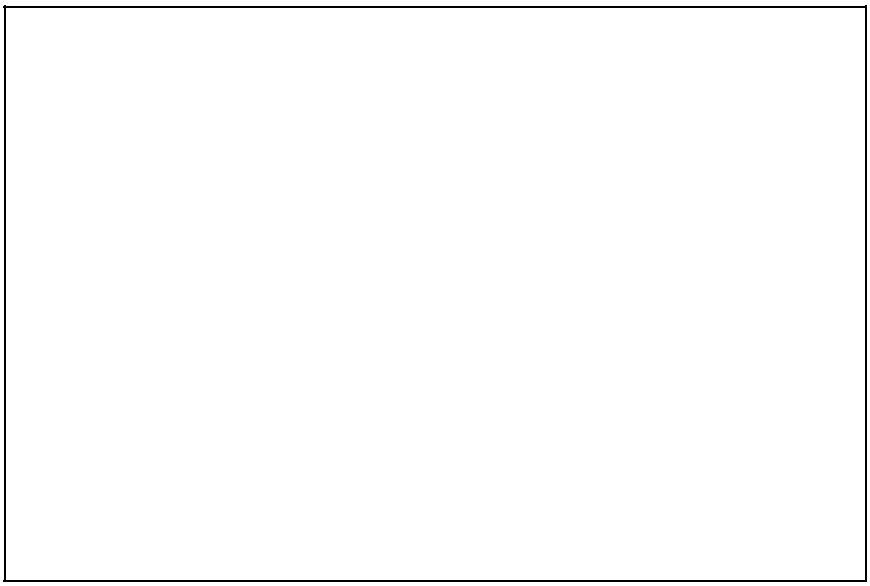


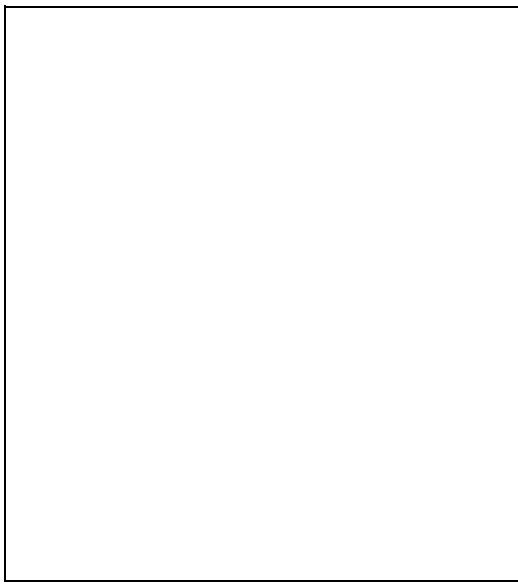
Writes/bob: two-node/50KiB



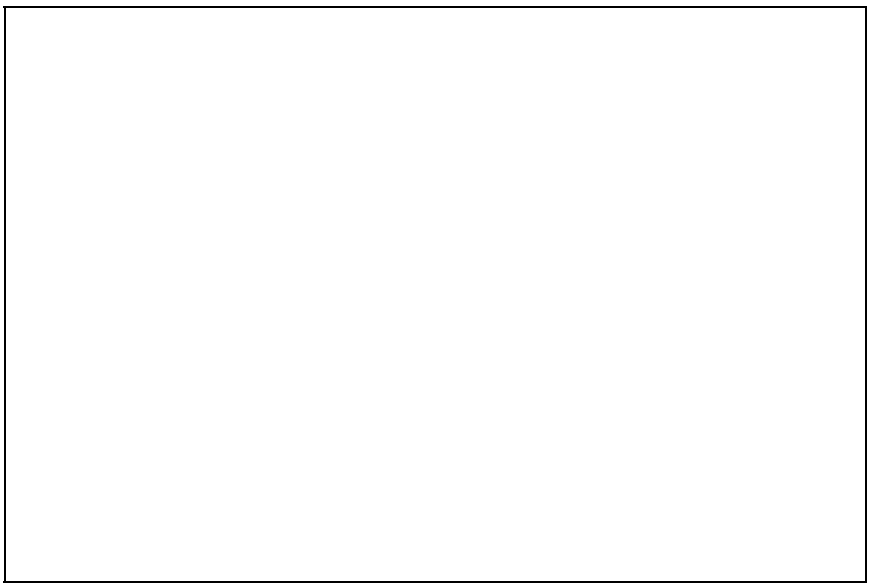


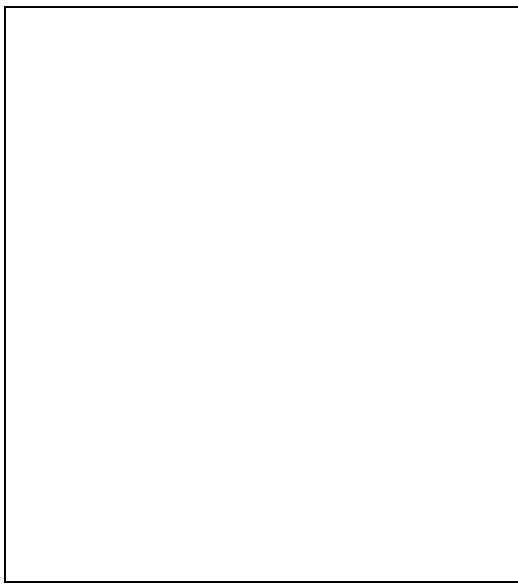
Writes/minio: one-node/100MiB



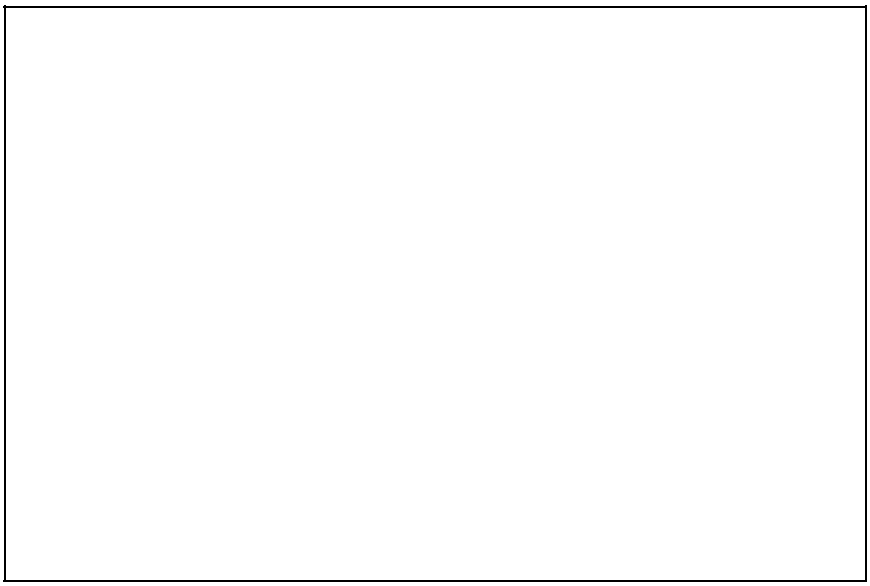


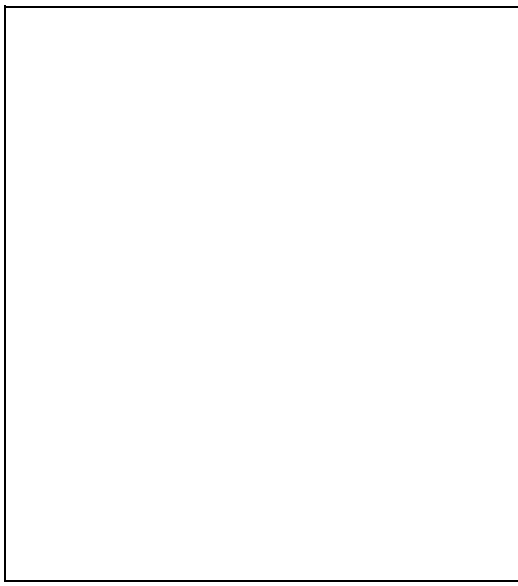
Writes/minio: one-node/1KiB



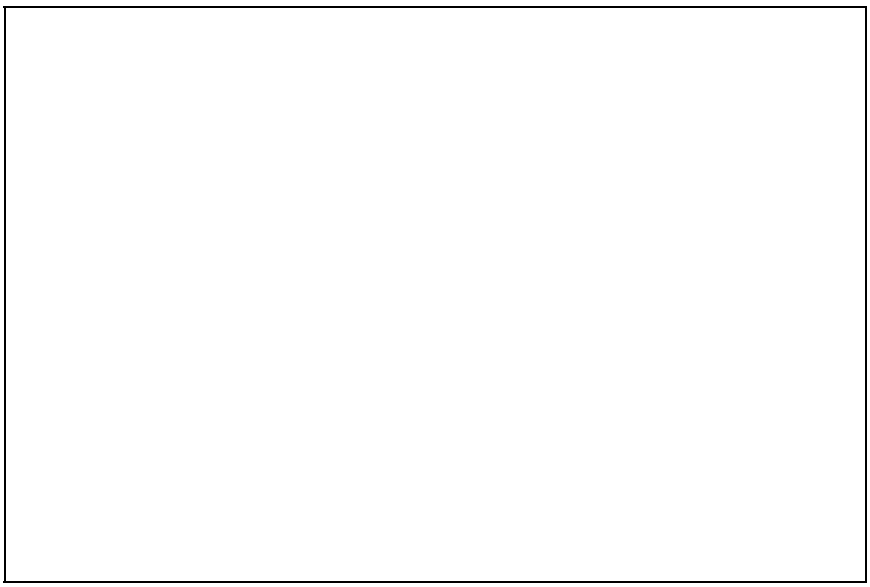


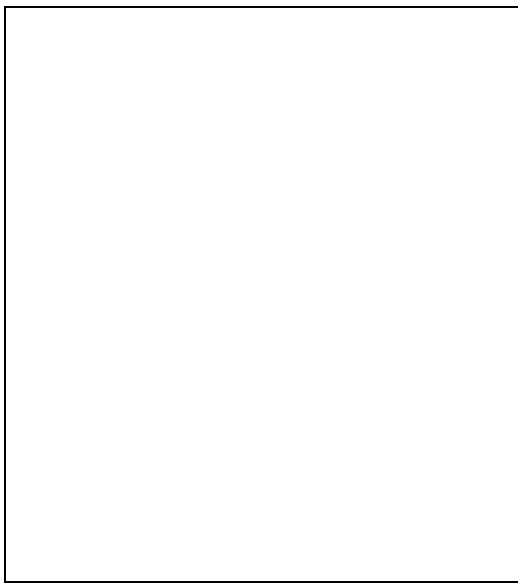
Writes/minio: one-node/1MiB



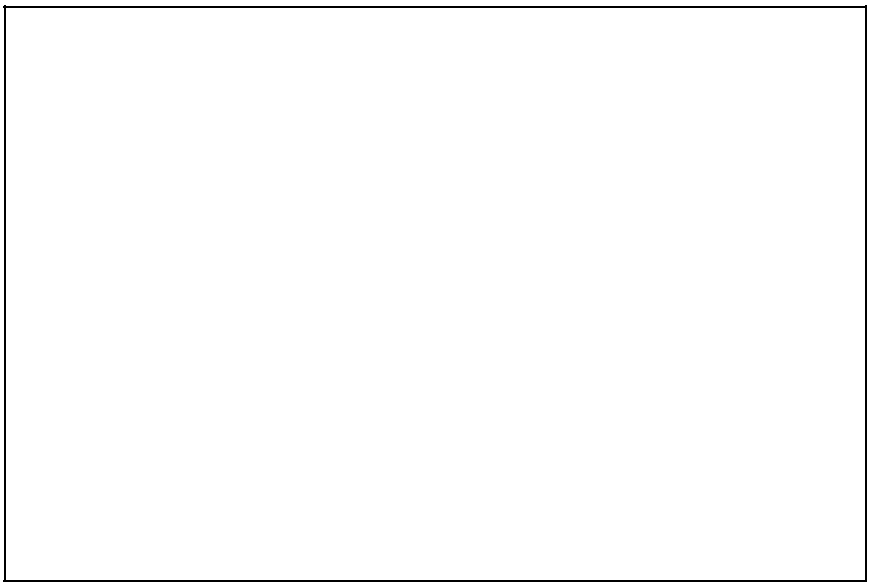


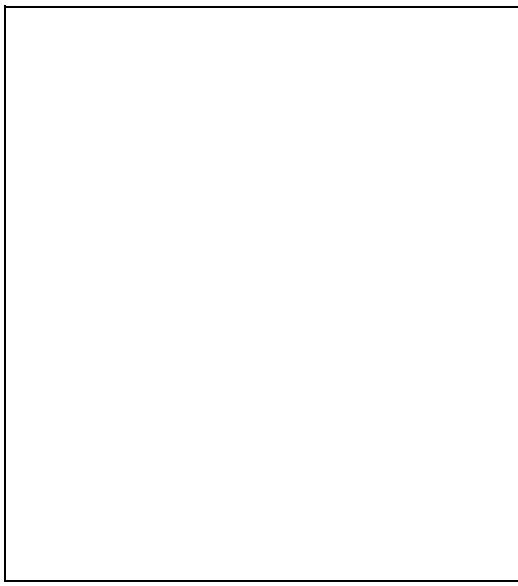
Writes/minio: one-node/50KiB



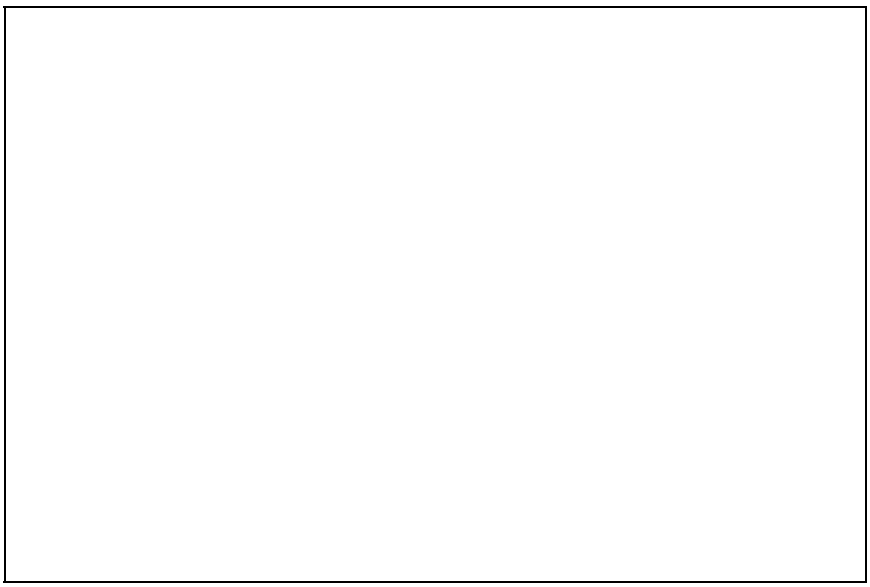


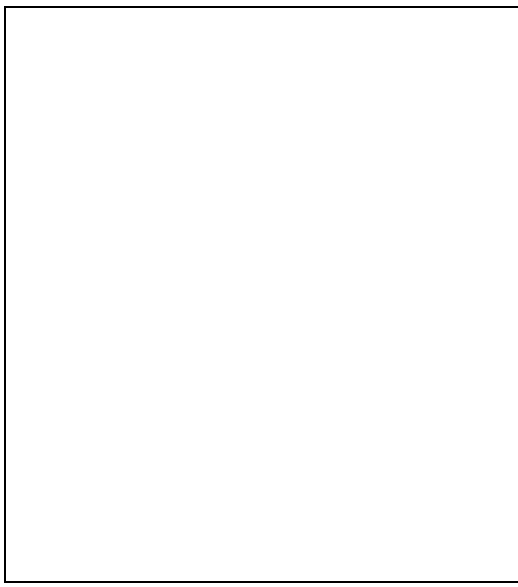
Writes/minio: two-node/100MiB



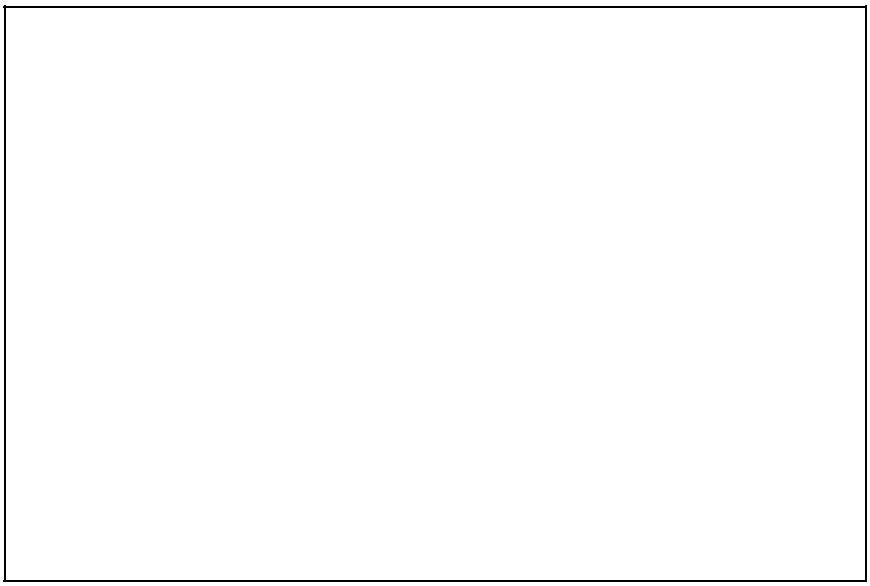


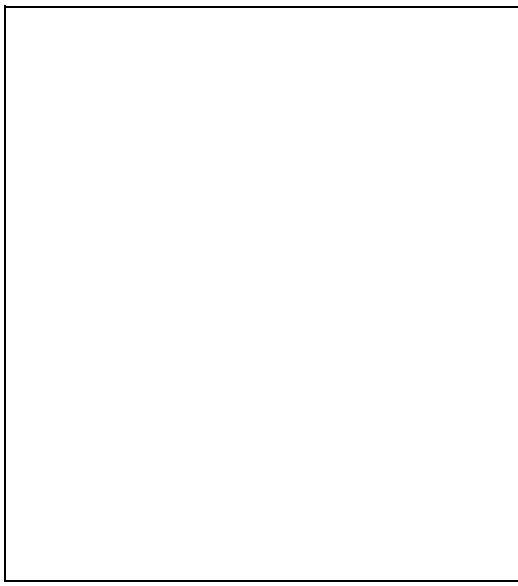
Writes/minio: two-node/1KiB



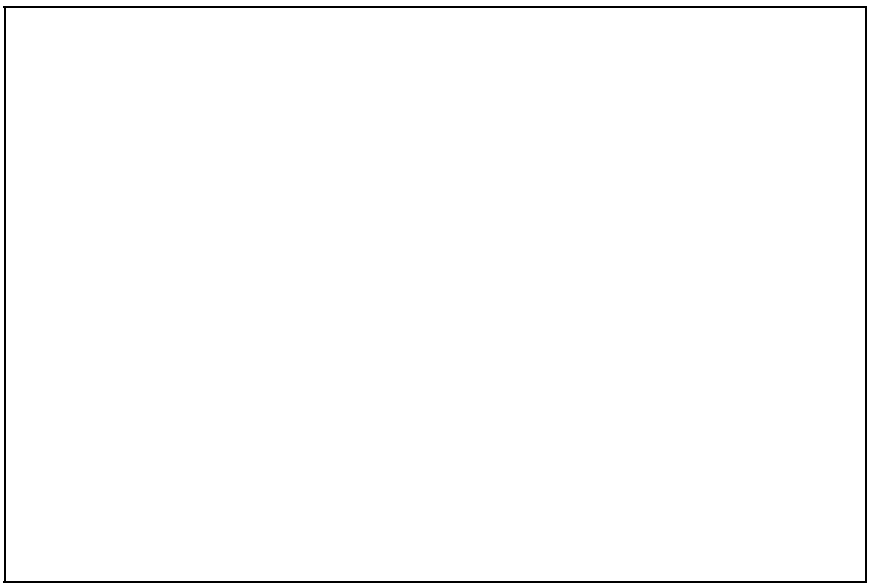


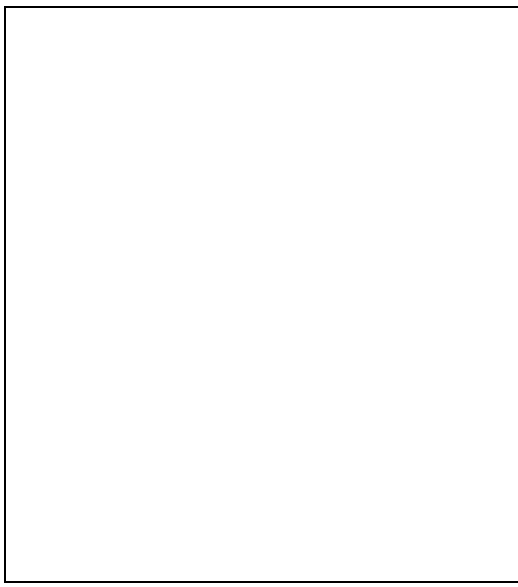
Writes/minio: two-node/1MiB





Writes/minio: two-node/50KiB





This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.

Criterion.rs Benchmark Index

See individual benchmark pages below for more details.

- Reads

	bob: one-node	bob: two-node	minio: one-node	minio: two-node
0B	Reads/minio: one-node/0B #4	Reads/minio: two-node/0B		
100MiB	Reads/bob: one-node/100MiB	Reads/bob: two-node/100MiB	Reads/minio: one-node/100MiB	Reads/minio: two-node/100MiB
1KiB	Reads/bob: one-node/1KiB	Reads/bob: two-node/1KiB	Reads/minio: one-node/1KiB	Reads/minio: two-node/1KiB
1MiB	Reads/bob: one-node/1MiB	Reads/bob: two-node/1MiB	Reads/minio: one-node/1MiB	Reads/minio: two-node/1MiB
50KiB	Reads/bob: one-node/50KiB	Reads/bob: two-node/50KiB	Reads/minio: one-node/50KiB	Reads/minio: two-node/50KiB

- Writes

	bob: one-node	bob: two-node	minio: one-node	minio: two-node
100MiB	Writes/bob: one-node/100MiB	Writes/bob: two-node/100MiB	Writes/minio: one-node/100MiB	Writes/minio: two-node/100MiB
1KiB	Writes/bob: one-node/1KiB	Writes/bob: two-node/1KiB	Writes/minio: one-node/1KiB	Writes/minio: two-node/1KiB
1MiB	Writes/bob: one-node/1MiB	Writes/bob: two-node/1MiB	Writes/minio: one-node/1MiB	Writes/minio: two-node/1MiB
50KiB	Writes/bob: one-node/50KiB	Writes/bob: two-node/50KiB	Writes/minio: one-node/50KiB	Writes/minio: two-node/50KiB

This report was generated by Criterion.rs, a statistics-driven benchmarking library in Rust.