



[www.phoronix-test-suite.com](http://www.phoronix-test-suite.com)

## native

Docker testing on Ubuntu 20.10 via the Phoronix Test Suite.

### Automated Executive Summary

*2 x 275GB Virtual Disk had the most wins, coming in first place for 86% of the tests.*

*Based on the geometric mean of all complete results, the fastest (2 x 275GB Virtual Disk) was 1.712x the speed of the slowest (lambda-cntr).*

*The results with the greatest spread from best to worst included:*

*Compile Bench (Test: Initial Create) at 28.729x*

*Compile Bench (Test: Compile) at 11.295x*

*Unpacking The Linux Kernel (linux-4.15.tar.xz) at 9.627x*

*IOzone (Record Size: 4Kb - File Size: 8GB - Disk Test: Write Performance) at 2.461x*

*Threaded I/O Tester (Test: Write - Size Per Thread: 64MB - Thread Count: 16) at 1.437x*

*Threaded I/O Tester (Test: Read - Size Per Thread: 64MB - Thread Count: 16) at 1.29x*

*LevelDB (Benchmark: Fill Sync) at 1.224x*

*Gzip Compression (Linux Source Tree Archiving To .tar.gz) at 1.213x*

*LevelDB (Benchmark: Fill Sync) at 1.213x*

*LevelDB (Benchmark: Overwrite) at 1.102x.*

## Test Systems:

### 2 x 275GB Virtual Disk

Processor: Intel Core i5-8305G (2 Cores / 4 Threads), Memory: 3072MB, Disk: 2 x 275GB Virtual Disk

OS: Ubuntu 20.10, Kernel: 4.19.84-microsoft-standard+ (x86\_64), Compiler: GCC 10.3.0, File-System: overlayfs, Screen Resolution: 1024x768, System Layer: Docker

Kernel Notes: Transparent Huge Pages: madvise  
 Compiler Notes: --build=x86\_64-linux-gnu --disable-vtable-verify --disable-werror --enable-bootstrap --enable-checking=release --enable-clocale=gnu --enable-default-pie --enable-gnu-unique-object --enable-languages=c,ada,c++,go,brig,d,fortran,objc,obj-c++,m2 --enable-libphobos-checking=release --enable-libstdcxx-debug --enable-libstdcxx-time=yes --enable-link-mutex --enable-multiarch --enable-multilib --enable-nls --enable-objc-gc=auto --enable-offload-targets=nvptx-none=/build/gcc-10-poYruo/gcc-10-10.3.0/debian/tmp-nvptx/usr,amdgcn-amdhsa=/build/gcc-10-poYruo/gcc-10-10.3.0/debian/tmp-gcn/usr,hsa --enable-plugin --enable-shared --enable-threads=posix --host=x86\_64-linux-gnu --program-prefix=x86\_64-linux-gnu- --target=x86\_64-linux-gnu --with-abi=m64 --with-arch-32=i686 --with-build-config=bootstrap-lto-lean --with-default-libstdcxx-abi=new --with-gcc-major-version-only --with-multilib-list=m32,m64,mx32 --with-target-system-zlib=auto --with-tune=generic --without-cuda-driver -v  
 Processor Notes: Scaling Governor: intel\_cpfufreq ondemand - CPU Microcode: 0x5003102

Disk Scheduler Notes: MQ-DEADLINE

Python Notes: Python 2.7.18 + Python 3.8.10

Security Notes: itlb\_multithit: KVM: Mitigation of VMX disabled + l1tf: Not affected + mds: Not affected + meltdown: Not affected + spec\_store\_bypass: Mitigation of SSB disabled via prctl and seccomp + spectre\_v1: Mitigation of usercopy/swaps barriers and \_\_user pointer sanitization + spectre\_v2: Mitigation of Enhanced IBRS IPBP: conditional RSB filling + srbs: Not affected + tsx\_async\_abort: Mitigation of TSX disabled

### lambda-cntr

Processor: Intel Core i5-8305G (2 Cores / 4 Threads), Memory: 3072MB, Disk: 2 x 275GB Virtual Disk

OS: Ubuntu 20.10, Kernel: 4.19.84-microsoft-standard+ (x86\_64), Compiler: GCC 10.3.0, File-System: fuseblk, Screen Resolution: 1024x768, System Layer: Docker

Kernel Notes: Transparent Huge Pages: madvise  
 Compiler Notes: --build=x86\_64-linux-gnu --disable-vtable-verify --disable-werror --enable-bootstrap --enable-checking=release --enable-clocale=gnu --enable-default-pie --enable-gnu-unique-object --enable-languages=c,ada,c++,go,brig,d,fortran,objc,obj-c++,m2 --enable-libphobos-checking=release --enable-libstdcxx-debug --enable-libstdcxx-time=yes --enable-link-mutex --enable-multiarch --enable-multilib --enable-nls --enable-objc-gc=auto --enable-offload-targets=nvptx-none=/build/gcc-10-poYruo/gcc-10-10.3.0/debian/tmp-nvptx/usr,amdgcn-amdhsa=/build/gcc-10-poYruo/gcc-10-10.3.0/debian/tmp-gcn/usr,hsa --enable-plugin --enable-shared --enable-threads=posix --host=x86\_64-linux-gnu --program-prefix=x86\_64-linux-gnu- --target=x86\_64-linux-gnu --with-abi=m64 --with-arch-32=i686 --with-build-config=bootstrap-lto-lean --with-default-libstdcxx-abi=new --with-gcc-major-version-only --with-multilib-list=m32,m64,mx32 --with-target-system-zlib=auto --with-tune=generic --without-cuda-driver -v  
 Disk Notes: MQ-DEADLINE / allow\_other,default\_permissions,group\_id=0,relatime,rw,user\_id=0 / Block Size: 4096  
 Processor Notes: Scaling Governor: intel\_cpfufreq ondemand - CPU Microcode: 0x5003102  
 Python Notes: Python 2.7.18 + Python 3.8.10  
 Security Notes: itlb\_multithit: KVM: Mitigation of VMX disabled + l1tf: Not affected + mds: Not affected + meltdown: Not affected + spec\_store\_bypass: Mitigation of SSB disabled via prctl and seccomp + spectre\_v1: Mitigation of usercopy/swaps barriers and \_\_user pointer sanitization + spectre\_v2: Mitigation of Enhanced IBRS IPBP: conditional RSB filling + srbs: Not affected + tsx\_async\_abort: Mitigation of TSX disabled

	2 x 275GB Virtual Disk	lambda-cntr
<b>LevelDB - Hot Read (us/Op)</b>	<b>52.960</b>	<b>53.911</b>
Normalized	100%	98.24%
Standard Deviation	5.2%	3.2%
<b>LevelDB - Fill Sync (MB/s)</b>	<b>6</b>	<b>4.9</b>
Normalized	100%	81.67%
Standard Deviation		1.2%
<b>LevelDB - Fill Sync (us/Op)</b>	<b>1024</b>	<b>1242</b>
Normalized	100%	82.47%
Standard Deviation	0.8%	1.2%
<b>LevelDB - Overwrite (MB/s)</b>	<b>10.8</b>	<b>9.8</b>

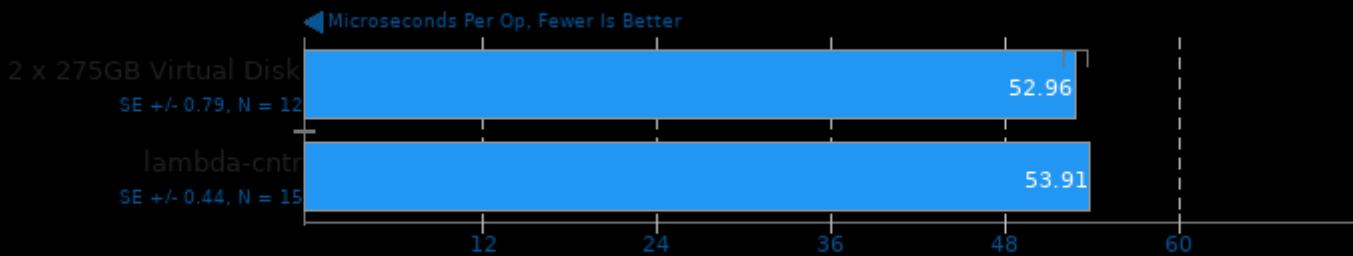
## native

	Normalized	100%	90.74%
	Standard Deviation	0.5%	0%
<b>LevelDB - Overwrite (us/Op)</b>	<b>574.525</b>	<b>632.700</b>	
	Normalized	100%	90.81%
	Standard Deviation	0.2%	0.5%
<b>LevelDB - Rand Fill (MB/s)</b>	<b>10.7</b>	<b>9.8</b>	
	Normalized	100%	91.59%
	Standard Deviation	0%	0%
<b>LevelDB - Rand Fill (us/Op)</b>	<b>577.638</b>	<b>630.760</b>	
	Normalized	100%	91.58%
	Standard Deviation	0.4%	0.3%
<b>LevelDB - Rand Read (us/Op)</b>	<b>53.322</b>	<b>54.119</b>	
	Normalized	100%	98.53%
	Standard Deviation	5.8%	6%
<b>LevelDB - Seek Rand (us/Op)</b>	<b>61.355</b>	<b>62.056</b>	
	Normalized	100%	98.87%
	Standard Deviation	0.6%	1.6%
<b>LevelDB - Rand Delete (us/Op)</b>	<b>555.659</b>	<b>598.903</b>	
	Normalized	100%	92.78%
	Standard Deviation	0.2%	0.4%
<b>LevelDB - Seq Fill (MB/s)</b>	<b>10.7</b>	<b>9.8</b>	
	Normalized	100%	91.59%
	Standard Deviation	0.9%	0.6%
<b>LevelDB - Seq Fill (us/Op)</b>	<b>576.761</b>	<b>632.882</b>	
	Normalized	100%	91.13%
	Standard Deviation	0.8%	0.5%
<b>IOzone - 4Kb - 8GB - Read Performance (MB/s)</b>	<b>5129</b>	<b>5371</b>	
	Normalized	95.5%	100%
	Standard Deviation	2.1%	5.8%
<b>IOzone - 4Kb - 8GB - Write Performance (MB/s)</b>	<b>18.58</b>	<b>7.55</b>	
	Normalized	100%	40.64%
	Standard Deviation	0.6%	0.6%
<b>Threaded I/O Tester - Read - 64MB - 16 (MB/s)</b>	<b>22980</b>	<b>29648</b>	
	Normalized	77.51%	100%
	Standard Deviation	1.2%	3.3%
<b>Threaded I/O Tester - Write - 64MB - 16 (MB/s)</b>	<b>82.958</b>	<b>57.744</b>	
	Normalized	100%	69.61%
	Standard Deviation	2.2%	2.4%
<b>Threaded I/O Tester - Rand Read - 64MB - 16 (MB/s)</b>	<b>204519</b>	<b>229092</b>	
	Normalized	89.27%	100%
	Standard Deviation	26.4%	1.3%
<b>Threaded I/O Tester - Rand Write - 64MB - 16 (MB/s)</b>	<b>3238</b>	<b>1796</b>	
	Normalized	100%	55.45%
	Standard Deviation	7%	0.5%
<b>Compile Bench - Compile (MB/s)</b>	<b>1880</b>	<b>166.40</b>	
	Normalized	100%	8.85%
	Standard Deviation	3.2%	2.3%
<b>Compile Bench - Initial Create (MB/s)</b>	<b>464.54</b>	<b>16.17</b>	
	Normalized	100%	3.48%
	Standard Deviation	4.9%	1.9%
<b>Compile Bench - Read Compiled Tree (MB/s)</b>	<b>815.61</b>	<b>161.33</b>	
	Normalized	100%	19.78%
	Standard Deviation	14.3%	11.5%
<b>Unpacking The Linux Kernel - linux-4.15.tar.xz (sec)</b>	<b>7.195</b>	<b>69.265</b>	
	Normalized	100%	10.39%

Standard Deviation	0.7%	1%
<b>Gzip Compression - L.S.T.A.T.t.g (sec)</b>	<b>43.364</b>	<b>52.584</b>
Normalized	100%	82.47%
Standard Deviation	0.1%	2.8%

## LevelDB 1.22

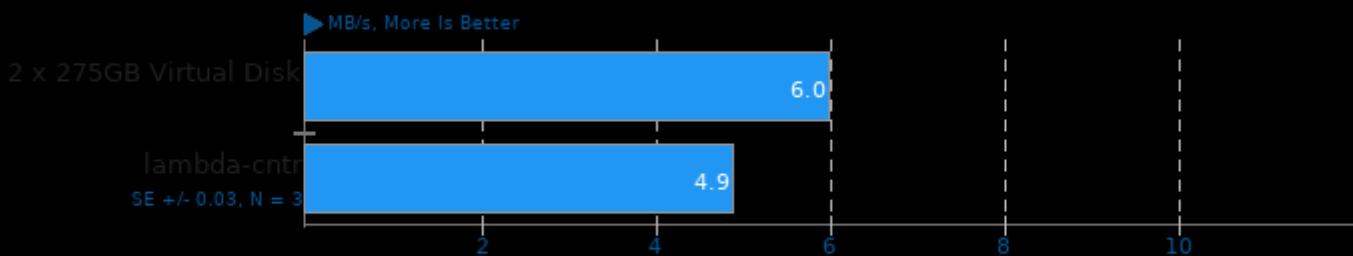
Benchmark: Hot Read



1. (CXX) g++ options: -O3 -lsnappy -lpthread

## LevelDB 1.22

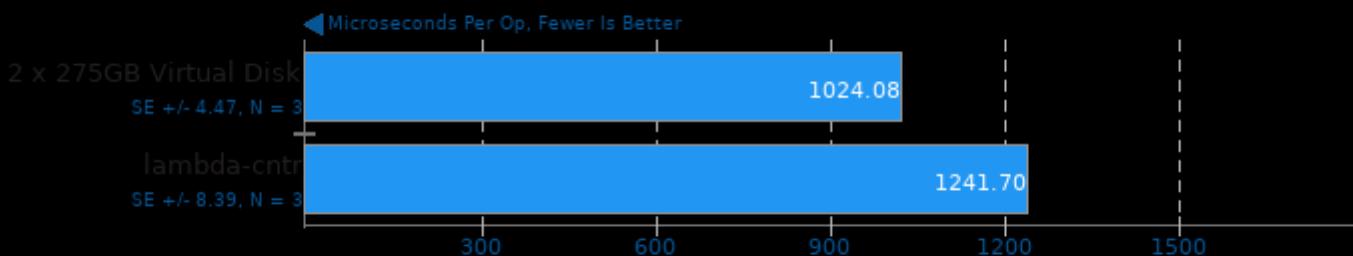
Benchmark: Fill Sync



1. (CXX) g++ options: -O3 -lsnappy -lpthread

## LevelDB 1.22

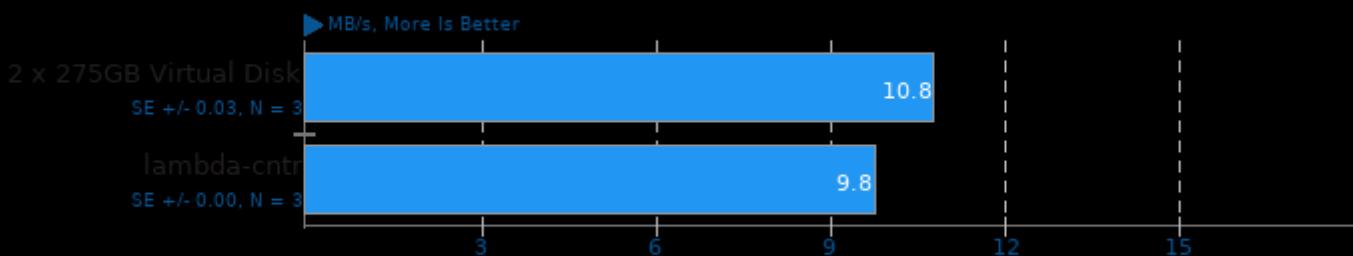
Benchmark: Fill Sync



1. (CXX) g++ options: -O3 -lsnappy -lpthread

## LevelDB 1.22

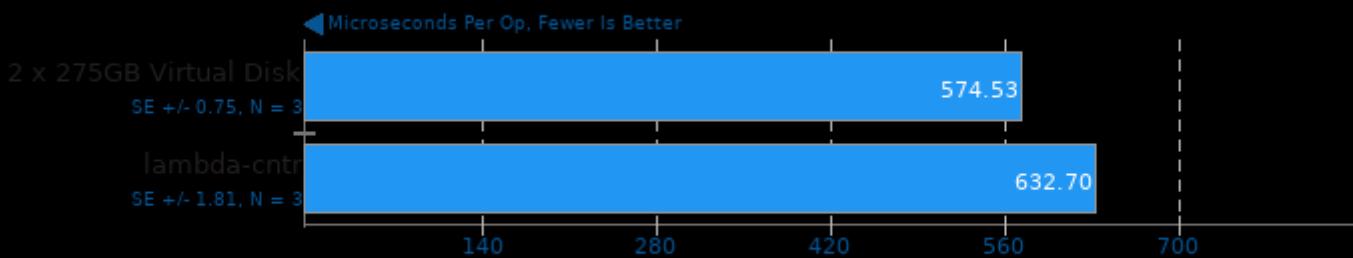
Benchmark: Overwrite



1. (CXX) g++ options: -O3 -lsnappy -lpthread

## LevelDB 1.22

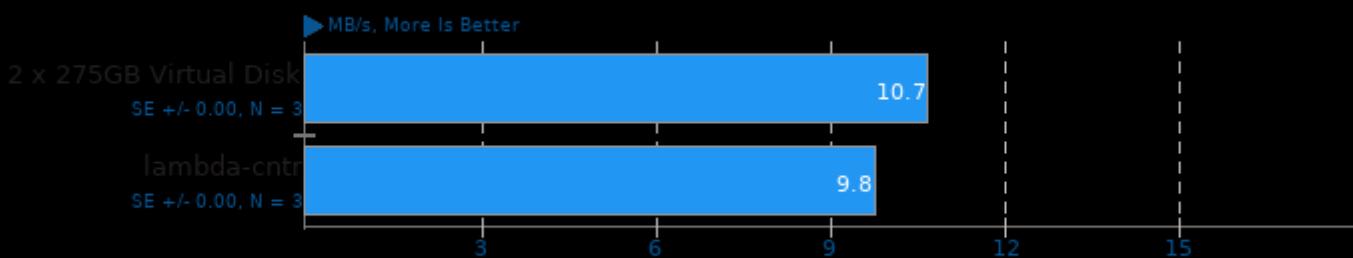
Benchmark: Overwrite



1. (CXX) g++ options: -O3 -lsnappy -lpthread

## LevelDB 1.22

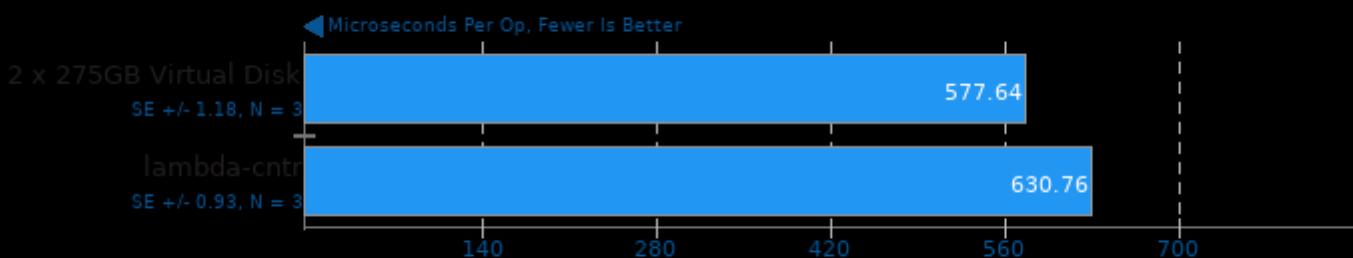
Benchmark: Random Fill



1. (CXX) g++ options: -O3 -lsnappy -lpthread

## LevelDB 1.22

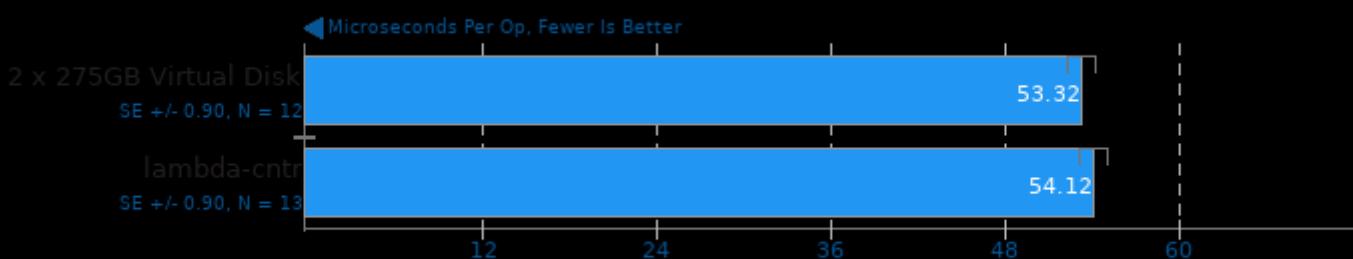
Benchmark: Random Fill



1. (CXX) g++ options: -O3 -lsnappy -lpthread

## LevelDB 1.22

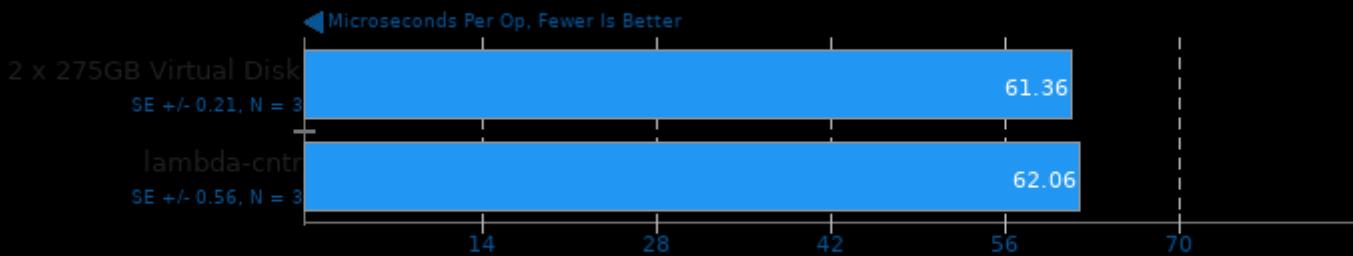
Benchmark: Random Read



1. (CXX) g++ options: -O3 -lsnappy -lpthread

## LevelDB 1.22

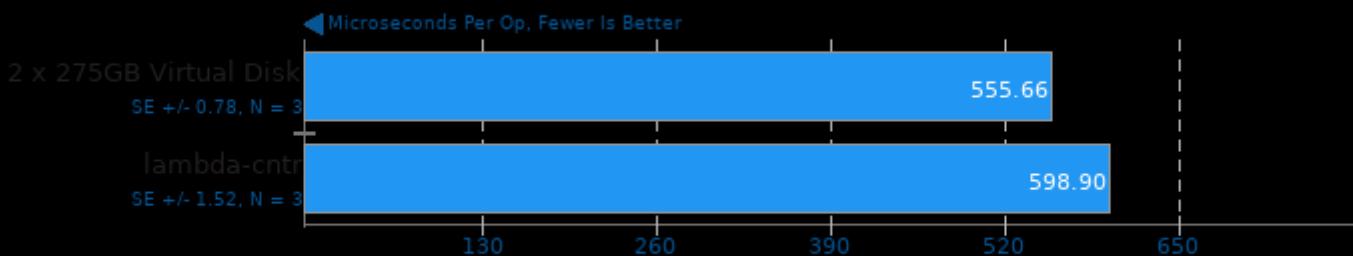
Benchmark: Seek Random



1. (CXX) g++ options: -O3 -lsnappy -lpthread

## LevelDB 1.22

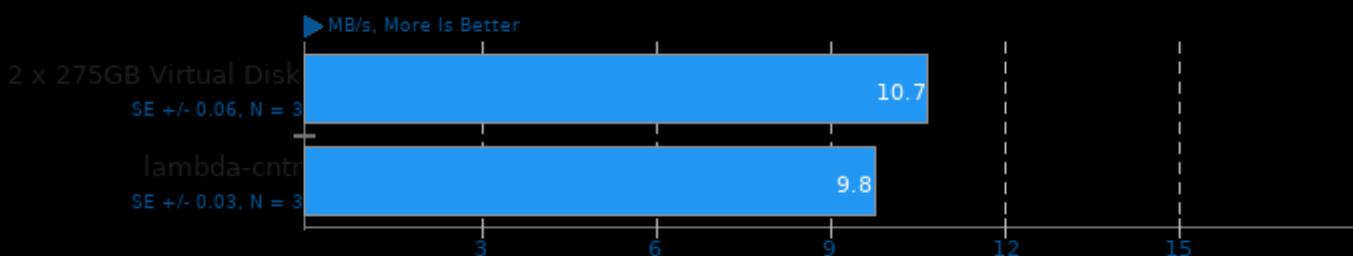
Benchmark: Random Delete



1. (CXX) g++ options: -O3 -lsnappy -lpthread

## LevelDB 1.22

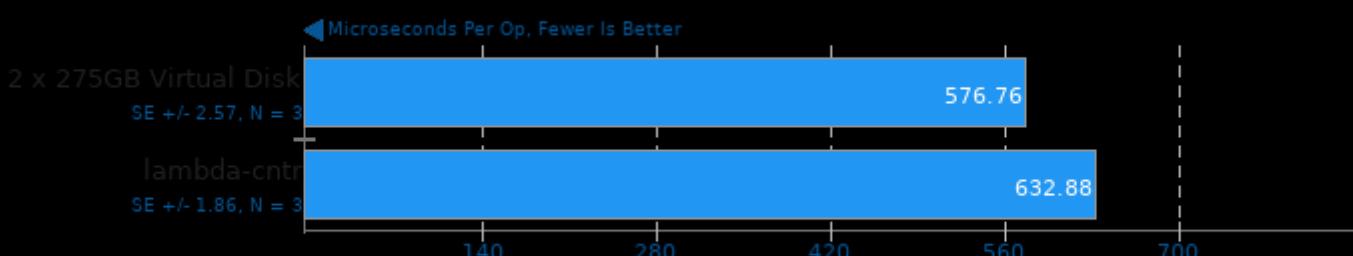
Benchmark: Sequential Fill



1. (CXX) g++ options: -O3 -lsnappy -lpthread

## LevelDB 1.22

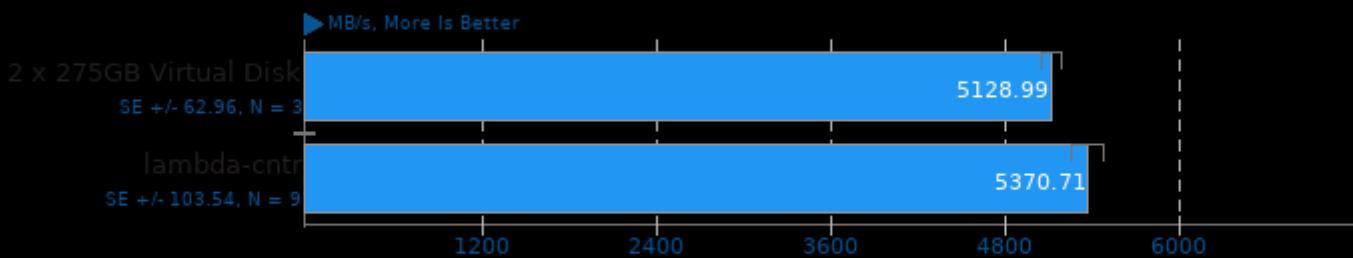
Benchmark: Sequential Fill



1. (CXX) g++ options: -O3 -lsnappy -lpthread

## IOzone 3.465

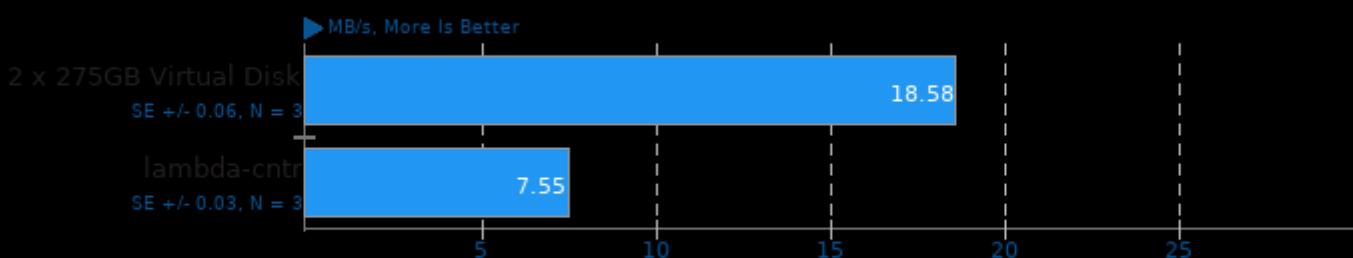
Record Size: 4Kb - File Size: 8GB - Disk Test: Read Performance



1. (CC) gcc options: -O3

## IOzone 3.465

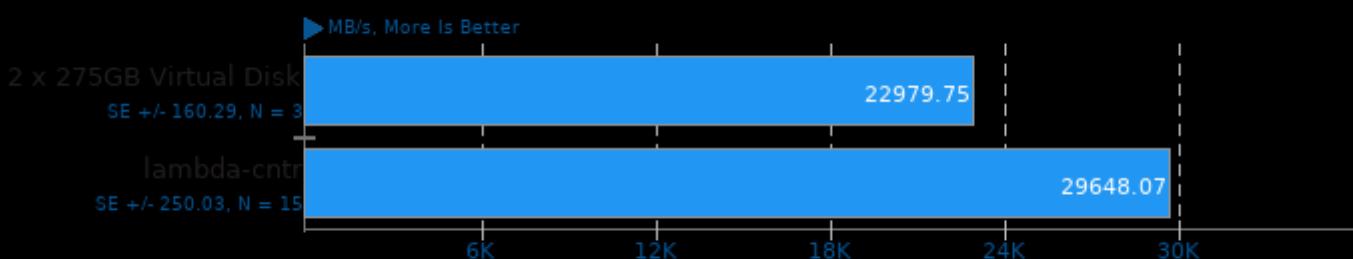
Record Size: 4Kb - File Size: 8GB - Disk Test: Write Performance



1. (CC) gcc options: -O3

## Threaded I/O Tester 20170503

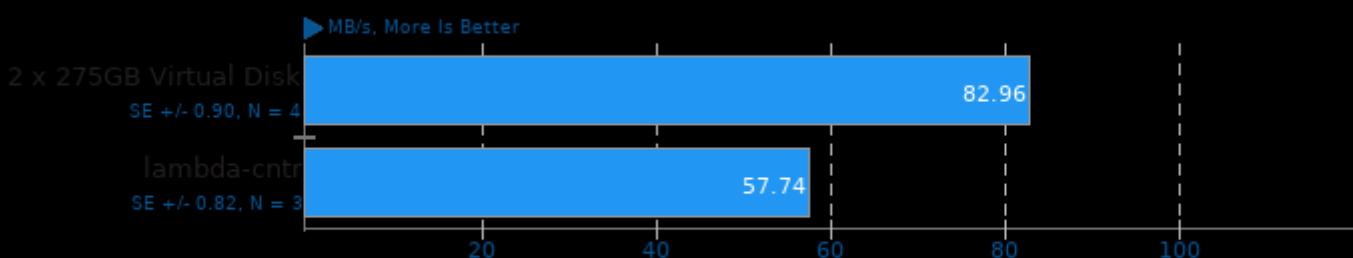
Test: Read - Size Per Thread: 64MB - Thread Count: 16



1. (CC) gcc options: -O2

## Threaded I/O Tester 20170503

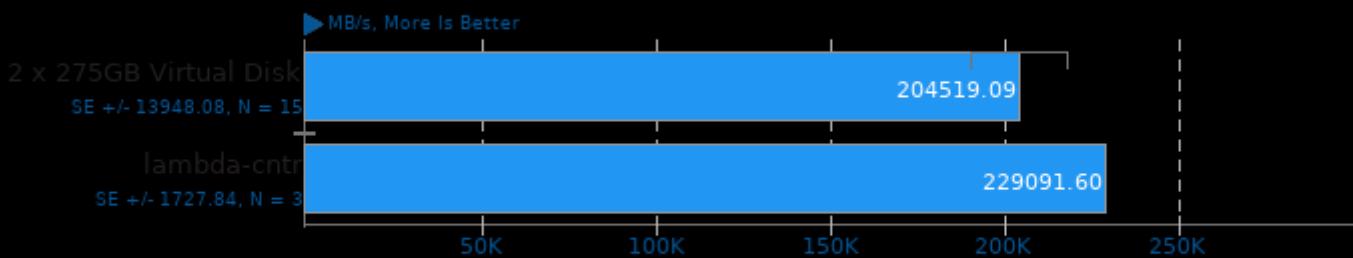
Test: Write - Size Per Thread: 64MB - Thread Count: 16



1. (CC) gcc options: -O2

**native****Threaded I/O Tester 20170503**

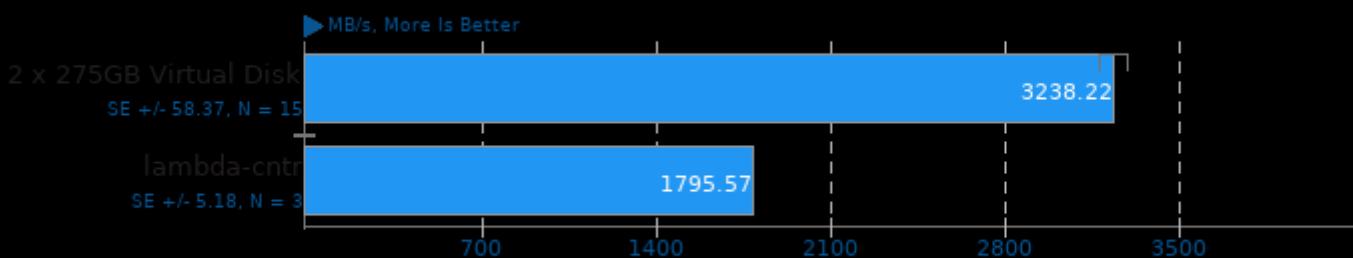
Test: Random Read - Size Per Thread: 64MB - Thread Count: 16



1. (CC) gcc options: -O2

**Threaded I/O Tester 20170503**

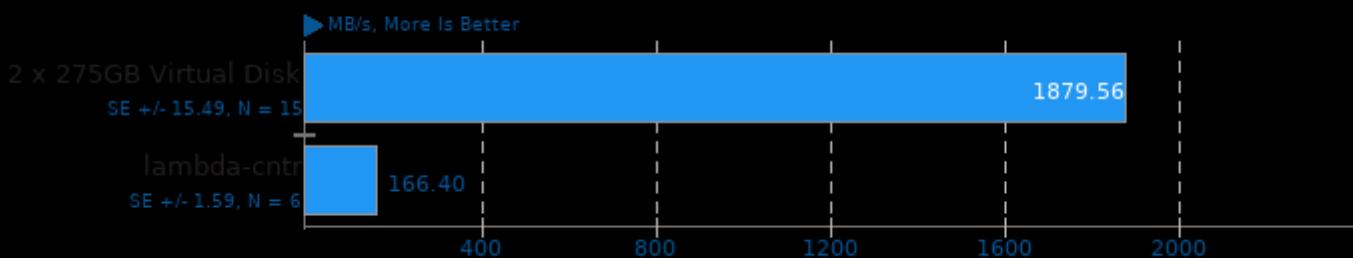
Test: Random Write - Size Per Thread: 64MB - Thread Count: 16



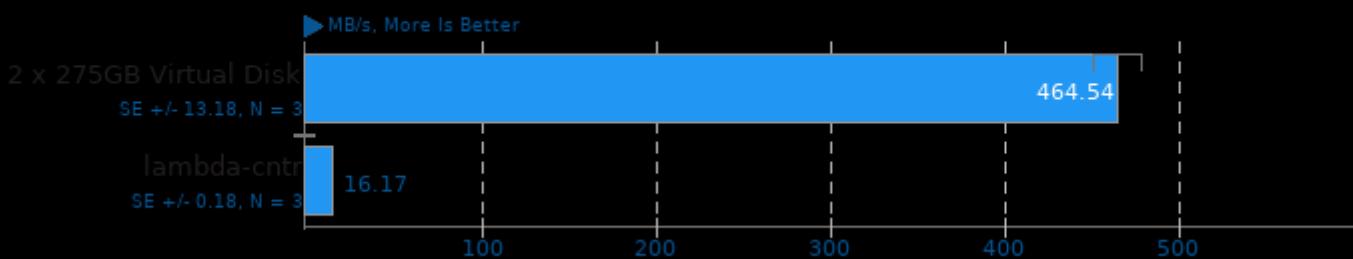
1. (CC) gcc options: -O2

**Compile Bench 0.6**

Test: Compile

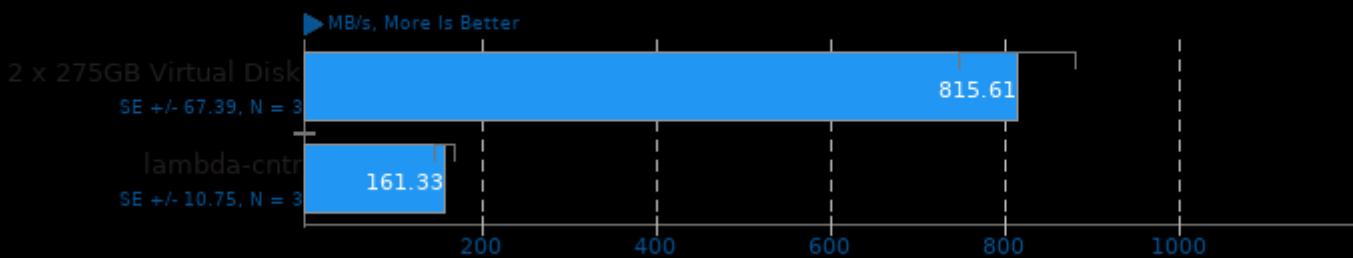
**Compile Bench 0.6**

Test: Initial Create



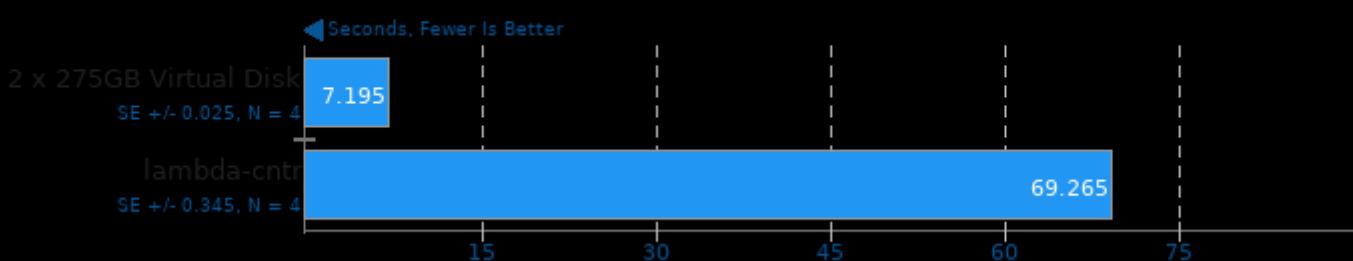
## Compile Bench 0.6

Test: Read Compiled Tree



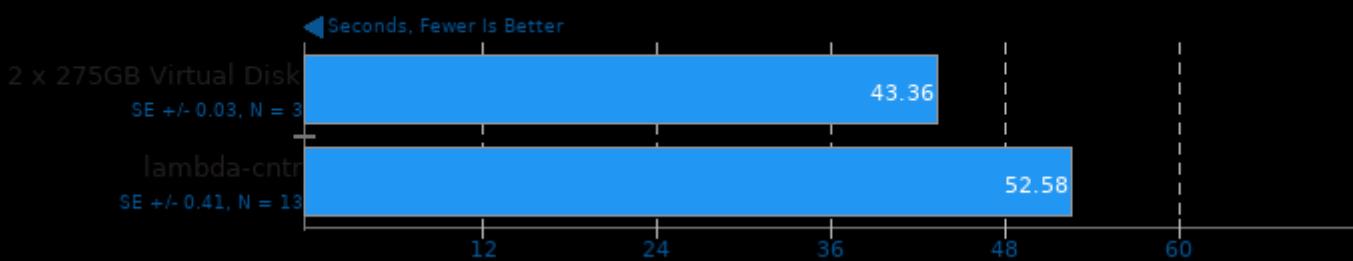
## Unpacking The Linux Kernel

linux-4.15.tar.xz

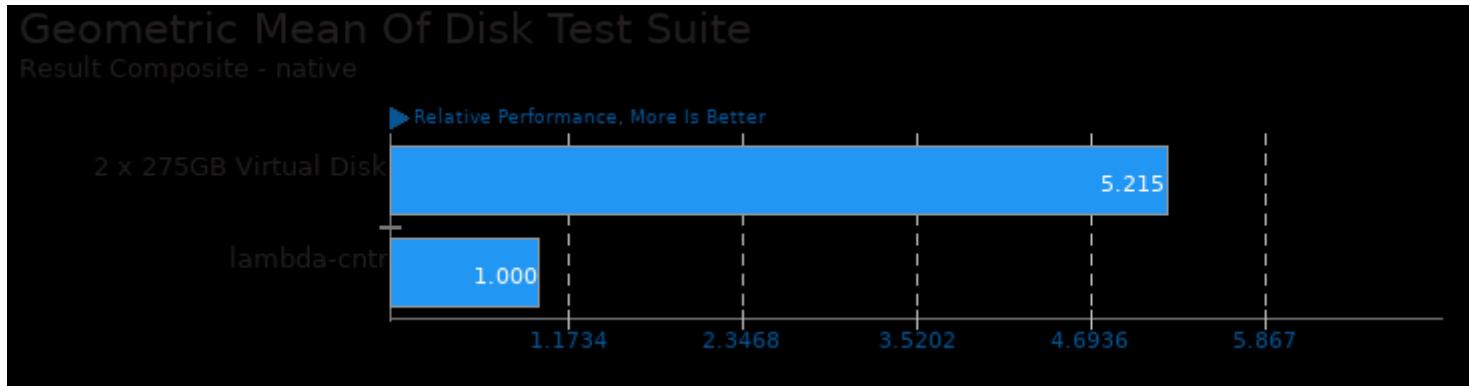


## Gzip Compression

Linux Source Tree Archiving To .tar.gz



These geometric means are based upon test groupings / test suites for this result file.



Geometric mean based upon tests: pts/compilebench and pts/iozone

*This file was automatically generated via the Phoronix Test Suite benchmarking software on Friday, 29 March 2024 08:46.*