

COEN 241 - ASSIGNMENT 1

Installation of QEMU and DOCKER

System Configurations

In this part of the report we will look at the process of installing the QEMU and on QEMU disk image. This will be used for the all the experiments in this assignment. The system configuration for this report as follows:



Download ISO - Ubuntu

Download the ubuntu iso from the following link:[Ubuntu 20.04_3 Server for ARM](https://raw.githubusercontent.com/Homebrew/install/master/install.sh) and place the iso file in the appropriate directory in ubuntu as we will use the iso file to create the image for the QEMU virtual machine.

1. Install homebrew using the following command:
 - b. `/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install.sh)"`
 - c. Add Homebrew to your PATH in `~/.zprofile`:
`echo 'eval "$(opt/homebrew/bin/brew shellenv)"' >> ~/.zprofile`
`eval "$(opt/homebrew/bin/brew shellenv)"`
 - d. After you've installed Homebrew, check that Homebrew is installed properly by typing the following command:
 - i. `brew doctor`

e. Type brew install qemu to install QEMU

Creating an QEMU image

Create the QEMU image where we install the VM. To create an image type the following command in the terminal:

```
qemu-img create ubuntu.img 15G -f qcow2
[ashokkumarrangababu@Ashoks-MacBook-Air build % qemu-img create ubuntu.img 15G -f]
  qcow2
Formatting 'ubuntu.img', fmt=qcow2 cluster_size=65536 extended_l2=off compression_type=zlib size=16106127360 lazy_refcounts=off refcount_bits=16
ashokkumarrangababu@Ashoks-MacBook-Air build %
```

We are creating an image named ubuntu with 15GB of space and -f denotes the file format, and we have chosen the file format as qcow2.

Installing the VM

To install the VM, run the following command on a terminal.

```
qemu-system-aarch64 \
-accel hvf -cpu cortex-a57 -M virt,highmem=off -m 2G \
-smp 2 \
-drive file=/opt/homebrew/Cellar/qemu/7.1.0/share/qemu/edk2-aarch64-code.fd,if=pflash,format=raw,readonly=on \
-drive if=none,file=ubuntu.img,format=qcow2,id=hd0 \
-device virtio-blk-device,drive=hd0,serial="trial_2" \
-device virtio-net-device,netdev=net0 \
-netdev user,id=net0 \
-vga none -device ramfb \
-cdrom /Users/ashokkumarrangababu/Downloads/ubuntu-20.04.5-live-server-arm64.iso \
-device usb-ehci -device usb-kbd -device usb-mouse -usb -nographic
```

Once the command runs successfully, follow the instructions on the screen and we have installed the QEMU Ubuntu VM image in the system.

To run the image, run the above command without the -cdrom option.

Note: The following options will give the detailed explanation of the qemu-system-aarch64.

- a. -m — Denotes memory, gives the allocation for the RAM memory for the VM. We have used 2GB of memory in this assignment.
- b. -smp – Denotes the number of cores that can be allocated for the VM. we have allocated 2 cores for our VM.
- c. -accel – Denotes h/w accelerations like kvm

When we change the configuration of smp to one, we observe the change in no.of cores per socket modify from 2 to 1.

Before smp change:

```
paddy@paddy:~$ lscpu
Architecture:          aarch64
CPU op-mode(s):        64-bit
Byte Order:            Little Endian
CPU(s):                2
On-line CPU(s) list:   0,1
Thread(s) per core:    1
Core(s) per socket:    2
Socket(s):             1
NUMA node(s):          1
Vendor ID:             ARM
Model:                 0
Model name:            Cortex-A57
Stepping:              r1p0
BogoMIPS:              48.00
NUMA node0 CPU(s):     0,1
Vulnerability Itlb multihit: Not affected
Vulnerability L1tf:     Not affected
Vulnerability Mds:      Not affected
Vulnerability Meltdown: Not affected
Vulnerability Mmio stale data: Not affected
Vulnerability Spec store bypass: Vulnerable
Vulnerability Spectre v1: Mitigation; __user pointer sanitization
Vulnerability Spectre v2: Mitigation; CSV2, BHB
Vulnerability Srbds:    Not affected
Vulnerability Tsx async abort: Not affected
Flags:                 fp asimd evtstrm aes pmull sha1 sha2 crc32 fphp
                        asimdh dp cpuid dit
paddy@paddy:~$
```

After smp change:

```
paddy@paddy:~$ lscpu
Architecture:          aarch64
CPU op-mode(s):        64-bit
Byte Order:            Little Endian
CPU(s):                1
On-line CPU(s) list:   0
Thread(s) per core:    1
Core(s) per socket:    1
[Socket(s):             1
NUMA node(s):          1
Vendor ID:             ARM
Model:                 0
Model name:            Cortex-A57
Stepping:              r1p0
BogoMIPS:              48.00
NUMA node0 CPU(s):     0
Vulnerability Itlb multihit: Not affected
Vulnerability L1tf:     Not affected
Vulnerability Mds:      Not affected
[Vulnerability Meltdown: Not affected
Vulnerability Mmio stale data: Not affected
Vulnerability Spec store bypass: Vulnerable
Vulnerability Spectre v1: Mitigation; __user pointer sanitization
Vulnerability Spectre v2: Mitigation; CSV2, BHB
Vulnerability Srbds:    Not affected
Vulnerability Tsx async abort: Not affected
Flags:                 fp asimd evtstrm aes pmull sha1 sha2 crc32 fphp
                        asimdh dp cpuid dit
paddy@paddy:~$
```

UBUNTU VM Configuration

System:

Kernel: 5.4.0-128-generic aarch64 bits: 64 compiler: gcc v: 9.4.0

Console: N/A Distro: Ubuntu 20.04.5 LTS (Focal Fossa)

Machine:

Type: Qemu System: QEMU product: QEMU Virtual Machine v: virt-7.1
serial: <filter>
Mobo: N/A model: N/A serial: N/A UEFI: EFI Development Kit II / OVMF
v: 0.0.0 date: 02/06/2015

CPU:

Topology: Dual Core model: N/A bits: 64 type: MCP arch: ARMv8
features: Use -f option to see features bogomips: 0
Speed: N/A min/max: N/A
Core speeds (MHz): No speed data found for 2 cores.

Graphics:

Message: No ARM data found for this feature.
Display: server: No display server data found. Headless machine?
tty: 80x24
Message: Advanced graphics data unavailable in console. Try -G --display

Audio:

Message: No ARM data found for this feature.

Network:

Message: No ARM data found for this feature.
IF-ID-1: eth0 state: up speed: -1 duplex: unknown mac: <filter>

Drives:

Local Storage: total: 15.00 GiB used: 4.55 GiB (30.4%)
ID-1: /dev/vda model: trial size: 15.00 GiB

Partition:

ID-1: / size: 9.75 GiB used: 4.44 GiB (45.6%) fs: ext4 dev: /dev/dm-0
ID-2: /boot size: 1.69 GiB used: 106.8 MiB (6.2%) fs: ext4 dev: /dev/vda2

Sensors:

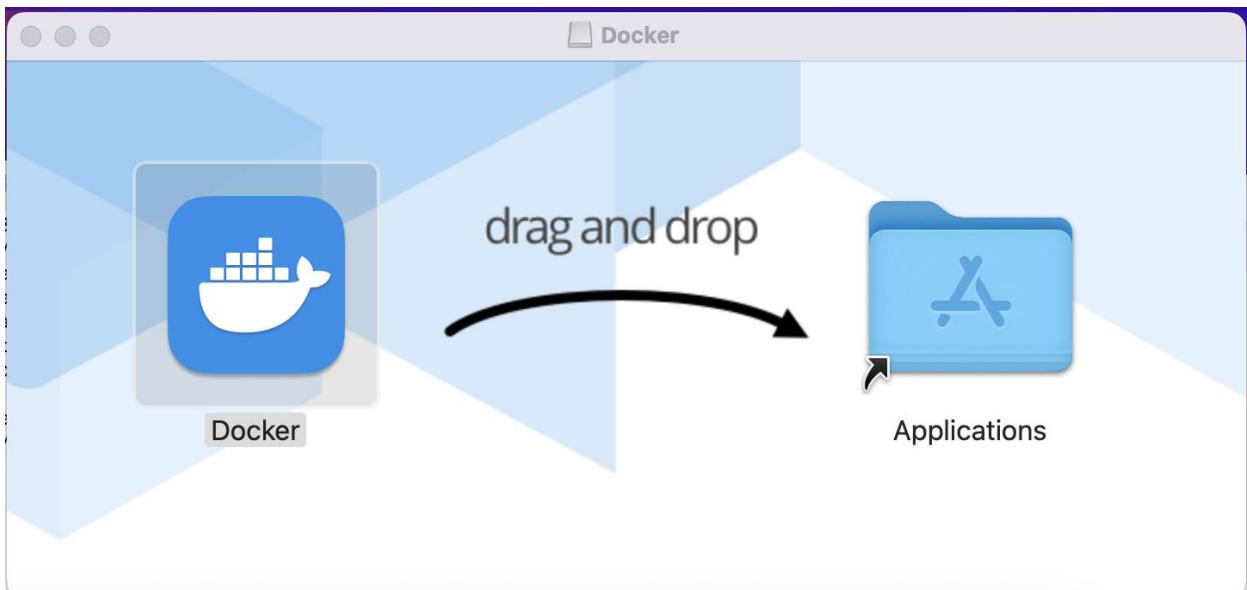
Message: No sensors data was found. Is sensors configured?

Info:

Processes: 95 Uptime: 1h 48m Memory: 1.93 GiB used: 204.0 MiB (10.3%)
Init: systemd runlevel: 5 Compilers: gcc: N/A Shell: bash v: 5.0.17
inxi: 3.0.38

DOCKER Installation

1. Install Rosetta using the following command:
 - a. softwareupdate -install-rosetta
2. Download the docker image from the following link: <https://docs.docker.com/desktop/mac/apple-silicon/>
3. Run the installer by doing the following



4. To run the docker engine, start the Docker application from the application folder
5. To check if the docker is installed and running, run this command on the terminal: docker run hello-world

```
[ashokkumarrangababu@Ashoks-MacBook-Air ~ % docker run hello-world
docker: Error response from daemon: dial unix docker.raw.sock: connect: no such file or directory.
See 'docker run --help'.
[ashokkumarrangababu@Ashoks-MacBook-Air ~ % docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
7050e35b49f5: Pull complete
Digest: sha256:18a657d0cc1c7d0678a3fbea8b7eb4918bba25968d3e1b0adefbfa71caddbc346
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.
```

To generate this message, Docker took the following steps:
1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
(arm64v8)
3. The Docker daemon created a new container from that image which runs the executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
\$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
<https://hub.docker.com/>

For more examples and ideas, visit:
<https://docs.docker.com/get-started/>

6. Open the terminal and run the following command:
 - a. docker run -it --name ubuntu:latest
 - b. docker start ubuntu
 - c. docker exec -it ubuntu bash
 - d. apt-get update
 - e. apt-get -y install sysbench

DOCKER Commands

The following are few of the docker commands:

1. docker version

```
[ashokkumarrangababu@Ashoks-MacBook-Air ~ % docker version
Client:
  Cloud integration: v1.0.29
  Version:          20.10.17
  API version:      1.41
  Go version:       go1.17.11
  Git commit:       100c701
  Built:            Mon Jun  6 23:04:45 2022
  OS/Arch:          darwin/arm64
  Context:          default
  Experimental:    true

Server: Docker Desktop 4.12.0 (85629)
Engine:
  Version:          20.10.17
  API version:      1.41 (minimum version 1.12)
  Go version:       go1.17.11
  Git commit:       a89b842
  Built:            Mon Jun  6 23:01:01 2022
  OS/Arch:          linux/arm64
  Experimental:    false
containerd:
  Version:          1.6.8
  GitCommit:        9cd3357b7fd7218e4aec3eae239db1f68a5a6ec6
runc:
  Version:          1.1.4
  GitCommit:        v1.1.4-0-g5fd4c4d
docker-init:
  Version:          0.19.0
  GitCommit:        de40ad0
```

2. docker images

```
[ashokkumarrangababu@Ashoks-MacBook-Air ~ % docker images
REPOSITORY          TAG      IMAGE ID      CREATED     SIZE
ubuntu              latest   d63f752103bb  12 days ago  69.2MB
hello-world         latest   46331d942d63  7 months ago  9.14kB
csminpp/ubuntu-sysbench  latest   2787c5e16909  6 years ago  336MB
```

EXPERIMENTS

QEMU Testing

- CPU Performance

Scenario 1

The CPU is tested using the cpu-max-prime where the max prime number is found under the given limit.
Command used: CPU Time - sysbench --test=cpu --cpu-max-prime=10000 run

Case 1:

```
Test Case: 1
sysbench 1.0.18 (using system LuaJIT 2.1.0-beta3)

Running the test with following options:
Number of threads: 1
Initializing random number generator from current time

Prime numbers limit: 10000
Initializing worker threads...

Threads started!

CPU speed:
events per second: 11027.52

General statistics:
total time:          10.0002s
total number of events: 110284

Latency (ms):
min:                 0.09
avg:                 0.09
max:                 0.35
95th percentile:    0.09
sum:                9988.66

Threads fairness:
events (avg/stddev): 110284.0000/0.00
execution time (avg/stddev): 9.9887/0.00
```

Case 2:

```
Test Case: 2
sysbench 1.0.18 (using system LuaJIT 2.1.0-beta3)

Running the test with following options:
Number of threads: 1
Initializing random number generator from current time

Prime numbers limit: 10000
Initializing worker threads...

Threads started!

CPU speed:
events per second: 11067.66

General statistics:
total time:          10.0001s
total number of events: 110685

Latency (ms):
min:                 0.09
avg:                 0.09
max:                 0.15
95th percentile:    0.09
sum:                9989.24

Threads fairness:
events (avg/stddev): 110685.0000/0.00
execution time (avg/stddev): 9.9892/0.00
```

Case 3:

```
Test Case: 3
sysbench 1.0.18 (using system LuaJIT 2.1.0-beta3)

Running the test with following options:
Number of threads: 1
Initializing random number generator from current time

Prime numbers limit: 10000
Initializing worker threads...
Threads started!
CPU speed:
events per second: 11059.83

General statistics:
total time:          10.0002s
total number of events: 110608

Latency (ms):
min:                  0.09
avg:                  0.09
max:                  0.17
95th percentile:     0.09
sum:                 9989.95

Threads fairness:
events (avg/stddev): 110608.0000/0.00
execution time (avg/stddev): 9.9900/0.00
```

Case 4:

```
Test Case: 4
sysbench 1.0.18 (using system LuaJIT 2.1.0-beta3)

Running the test with following options:
Number of threads: 1
Initializing random number generator from current time

Prime numbers limit: 10000
Initializing worker threads...
Threads started!
CPU speed:
events per second: 11048.16

General statistics:
total time:          10.0002s
total number of events: 110491

Latency (ms):
min:                  0.09
avg:                  0.09
max:                  0.16
95th percentile:     0.09
sum:                 9990.41

Threads fairness:
events (avg/stddev): 110491.0000/0.00
execution time (avg/stddev): 9.9904/0.00
```

Case 5:

```

Test Case: 5
sysbench 1.0.18 (using system LuaJIT 2.1.0-beta3)

Running the test with following options:
Number of threads: 1
Initializing random number generator from current time

Prime numbers limit: 10000
Initializing worker threads...
Threads started!

CPU speed:
events per second: 11058.12

General statistics:
total time: 10.0002s
total number of events: 110590

Latency (ms):
min: 0.09
avg: 0.09
max: 0.15
95th percentile: 0.09
sum: 9990.20

Threads fairness:
events (avg/stddev): 110590.0000/0.00
execution time (avg/stddev): 9.9902/0.00

```

Scenario 2

Command used: CPU Time - sysbench --test=cpu --cpu-max-prime=30000 run

Case 1:

```

-----
QEMU CPU Test
-----
Test Case: 1
sysbench 1.0.18 (using system LuaJIT 2.1.0-beta3)

Running the test with following options:
Number of threads: 1
Initializing random number generator from current time

Prime numbers limit: 30000
Initializing worker threads...
Threads started!

CPU speed:
events per second: 2478.65

General statistics:
total time: 10.0003s
total number of events: 24789

Latency (ms):
min: 0.39
avg: 0.40
max: 1.03
95th percentile: 0.41
sum: 9995.25

Threads fairness:
events (avg/stddev): 24789.0000/0.00
execution time (avg/stddev): 9.9952/0.00

```

Case 2:

```
Test Case: 2
sysbench 1.0.18 (using system LuaJIT 2.1.0-beta3)

Running the test with following options:
Number of threads: 1
Initializing random number generator from current time

Prime numbers limit: 30000
Initializing worker threads...
Threads started!

CPU speed:
events per second: 2485.70

General statistics:
total time: 10.0002s
total number of events: 24859

Latency (ms):
min: 0.39
avg: 0.40
max: 0.65
95th percentile: 0.41
sum: 9996.47

Threads fairness:
events (avg/stddev): 24859.0000/0.00
execution time (avg/stddev): 9.9965/0.00
```

Case 3:

```
Test Case: 3
sysbench 1.0.18 (using system LuaJIT 2.1.0-beta3)

Running the test with following options:
Number of threads: 1
Initializing random number generator from current time

Prime numbers limit: 30000
Initializing worker threads...
Threads started!

CPU speed:
events per second: 2482.32

General statistics:
total time: 10.0005s
total number of events: 24826

Latency (ms):
min: 0.40
avg: 0.40
max: 0.51
95th percentile: 0.41
sum: 9997.53

Threads fairness:
events (avg/stddev): 24826.0000/0.00
execution time (avg/stddev): 9.9975/0.00
```

Case 4:

```
Test Case: 4
sysbench 1.0.18 (using system LuaJIT 2.1.0-beta3)

Running the test with following options:
Number of threads: 1
Initializing random number generator from current time

Prime numbers limit: 30000
Initializing worker threads...
Threads started!

CPU speed:
events per second: 2487.49

General statistics:
total time: 10.0002s
total number of events: 24877

Latency (ms):
min: 0.39
avg: 0.40
max: 0.56
95th percentile: 0.41
sum: 9997.34

Threads fairness:
events (avg/stddev): 24877.0000/0.00
execution time (avg/stddev): 9.9973/0.00
```

Case 5:

```
Test Case: 5
sysbench 1.0.18 (using system LuaJIT 2.1.0-beta3)

Running the test with following options:
Number of threads: 1
Initializing random number generator from current time

Prime numbers limit: 30000
Initializing worker threads...
Threads started!

CPU speed:
events per second: 2486.53

General statistics:
total time: 10.0005s
total number of events: 24868

Latency (ms):
min: 0.39
avg: 0.40
max: 0.51
95th percentile: 0.41
sum: 9996.44

Threads fairness:
events (avg/stddev): 24868.0000/0.00
execution time (avg/stddev): 9.9964/0.00
```

Scenario 3

Command used: CPU Time - sysbench --test=cpu --cpu-max-prime=50000 run

Case 1:

```
-----
QEMU CPU Test
-----
Test Case: 1
sysbench 1.0.18 (using system LuaJIT 2.1.0-beta3)

Running the test with following options:
Number of threads: 1
Initializing random number generator from current time

Prime numbers limit: 50000

Initializing worker threads...

Threads started!

CPU speed:
events per second: 1232.77

General statistics:
total time: 10.0005s
total number of events: 12329

Latency (ms):
min: 0.79
avg: 0.81
max: 2.49
95th percentile: 0.81
sum: 9996.82

Threads fairness:
events (avg/stddev): 12329.0000/0.00
execution time (avg/stddev): 9.9968/0.00
```

Case 2:

```
Test Case: 2
sysbench 1.0.18 (using system LuaJIT 2.1.0-beta3)

Running the test with following options:
Number of threads: 1
Initializing random number generator from current time

Prime numbers limit: 50000

Initializing worker threads...

Threads started!

CPU speed:
events per second: 1235.48

General statistics:
total time: 10.0003s
total number of events: 12356

Latency (ms):
min: 0.79
avg: 0.81
max: 1.01
95th percentile: 0.83
sum: 9996.58

Threads fairness:
events (avg/stddev): 12356.0000/0.00
execution time (avg/stddev): 9.9966/0.00
```

Case 3:

```
Test Case: 3
sysbench 1.0.18 (using system LuaJIT 2.1.0-beta3)

Running the test with following options:
Number of threads: 1
Initializing random number generator from current time

Prime numbers limit: 50000

Initializing worker threads...

Threads started!

CPU speed:
events per second: 1235.73

General statistics:
total time: 10.0007s
total number of events: 12359

Latency (ms):
min: 0.79
avg: 0.81
max: 1.03
95th percentile: 0.83
sum: 9998.89

Threads fairness:
events (avg/stddev): 12359.0000/0.00
execution time (avg/stddev): 9.9989/0.00
```

Case 4

```
Test Case: 4
sysbench 1.0.18 (using system LuaJIT 2.1.0-beta3)

Running the test with following options:
Number of threads: 1
Initializing random number generator from current time

Prime numbers limit: 50000

Initializing worker threads...

Threads started!

CPU speed:
    events per second: 1235.96

General statistics:
    total time:          10.0005s
    total number of events: 12361

Latency (ms):
    min:                0.79
    avg:                0.81
    max:                1.00
    95th percentile:   0.83
    sum:               9998.44

Threads fairness:
    events (avg/stddev): 12361.0000/0.00
    execution time (avg/stddev): 9.9984/0.00
```

Case 5:

```

Test Case: 5
sysbench 1.0.18 (using system LuaJIT 2.1.0-beta3)

Running the test with following options:
Number of threads: 1
Initializing random number generator from current time

Prime numbers limit: 50000

Initializing worker threads...

Threads started!

CPU speed:
events per second: 1234.19

General statistics:
total time: 10.0003s
total number of events: 12343

Latency (ms):
min: 0.79
avg: 0.81
max: 1.02
95th percentile: 0.83
sum: 9998.21

Threads fairness:
events (avg/stddev): 12343.0000/0.00
execution time (avg/stddev): 9.9982/0.00

```

- File I/O testing

This testing is to determine the read and write throughput of the system.

Scenario 1

Command used:

```

sysbench --test=fileio --file-total-size=2G --file-test-mode=rndrw --max-time=30 --max-requests=0 --file-extra-flags=direct prepare
sysbench --test=fileio --file-total-size=2G --file-test-mode=rndrw --max-time=30 --max-requests=0 --file-extra-flags=direct run
sleep 60
sysbench --test=fileio --file-total-size=2G --file-test-mode=rndrw --max-time=30 --max-requests=0 --file-extra-flags=direct cleanup

```

Case 1:

```
File operations:
  reads/s:          5043.13
  writes/s:         3362.09
  fsyncs/s:         10760.52

Throughput:
  read, MiB/s:      78.80
  written, MiB/s:   52.53

General statistics:
  total time:       30.0047s
  total number of events: 574943

Latency (ms):
  min:              0.02
  avg:              0.05
  max:              18.51
  95th percentile:  0.15
  sum:              29850.81

Threads fairness:
  events (avg/stddev): 574943.0000/0.00
  execution time (avg/stddev): 29.8508/0.00
```

Case 2:

Threads started!

```
File operations:
  reads/s:          4004.10
  writes/s:         2669.38
  fsyncs/s:         8544.35

Throughput:
  read, MiB/s:      62.56
  written, MiB/s:   41.71

General statistics:
  total time:       30.0056s
  total number of events: 456502

Latency (ms):
  min:              0.02
  avg:              0.07
  max:              24.20
  95th percentile:  0.15
  sum:              29870.67

Threads fairness:
  events (avg/stddev): 456502.0000/0.00
  execution time (avg/stddev): 29.8707/0.00
```

Case 3:
Threads started!

```
File operations:
  reads/s:          4378.65
  writes/s:         2919.12
  fsyncs/s:         9342.05

Throughput:
  read, MiB/s:      68.42
  written, MiB/s:   45.61

General statistics:
  total time:       30.0058s
  total number of events: 499171

Latency (ms):
  min:              0.02
  avg:              0.06
  max:              3.86
  95th percentile:  0.15
  sum:              29862.69

Threads fairness:
  events (avg/stddev): 499171.0000/0.00
  execution time (avg/stddev): 29.8627/0.00
```

Case 4:
Threads started!

```
File operations:
  reads/s:          4297.90
  writes/s:         2865.28
  fsyncs/s:         9171.56

Throughput:
  read, MiB/s:      67.15
  written, MiB/s:   44.77

General statistics:
  total time:       30.0052s
  total number of events: 490009

Latency (ms):
  min:              0.02
  avg:              0.06
  max:              5.87
  95th percentile:  0.15
  sum:              29866.33

Threads fairness:
  events (avg/stddev): 490009.0000/0.00
  execution time (avg/stddev): 29.8663/0.00
```

Case 5:

```
Threads started!
```

```
File operations:
  reads/s:          4311.24
  writes/s:         2874.16
  fsyncs/s:         9199.00

Throughput:
  read, MiB/s:      67.36
  written, MiB/s:   44.91

General statistics:
  total time:       30.0048s
  total number of events: 491491

Latency (ms):
  min:              0.02
  avg:              0.06
  max:              3.16
  95th percentile:  0.16
  sum:              29858.17

Threads fairness:
  events (avg/stddev): 491491.0000/0.00
  execution time (avg/stddev): 29.8582/0.00
```

Scenario 2

Command used:

```
sysbench --test=fileio --file-total-size=3G --file-test-mode=rndrw --max-time=30 --max-requests=0 --file-extra-flags=direct prepare
sysbench --test=fileio --file-total-size=3G --file-test-mode=rndrw --max-time=30 --max-requests=0 --file-extra-flags=direct run
sleep 60
sysbench --test=fileio --file-total-size=3G --file-test-mode=rndrw --max-time=30 --max-requests=0 --file-extra-flags=direct cleanup
```

Case 1:

```
Threads started!

File operations:
  reads/s:          4493.72
  writes/s:         2995.80
  fsyncs/s:         9589.65

Throughput:
  read, MiB/s:      70.21
  written, MiB/s:   46.81

General statistics:
  total time:       30.0052s
  total number of events: 512344

Latency (ms):
  min:              0.02
  avg:              0.06
  max:              22.31
  95th percentile:  0.15
  sum:              29863.38

Threads fairness:
  events (avg/stddev): 512344.0000/0.00
  execution time (avg/stddev): 29.8634/0.00
```

Case 2:

```
Threads started!

File operations:
  reads/s:          4202.02
  writes/s:         2801.33
  fsyncs/s:         8966.81

Throughput:
  read, MiB/s:      65.66
  written, MiB/s:   43.77

General statistics:
  total time:       30.0053s
  total number of events: 479069

Latency (ms):
  min:              0.02
  avg:              0.06
  max:              2.26
  95th percentile:  0.14
  sum:              29879.47

Threads fairness:
  events (avg/stddev): 479069.0000/0.00
  execution time (avg/stddev): 29.8795/0.00
```

Case 3:

Threads started!

```
File operations:
  reads/s:          4288.84
  writes/s:         2859.23
  fsyncs/s:         9150.17

Throughput:
  read, MiB/s:      67.01
  written, MiB/s:   44.68

General statistics:
  total time:       30.0055s
  total number of events: 488917

Latency (ms):
  min:              0.02
  avg:              0.06
  max:              2.71
  95th percentile:  0.14
  sum:              29878.70

Threads fairness:
  events (avg/stddev): 488917.0000/0.00
  execution time (avg/stddev): 29.8787/0.00
```

Case 4:

Threads started!

```
File operations:
  reads/s:          4490.51
  writes/s:         2993.65
  fsyncs/s:         9580.97

Throughput:
  read, MiB/s:      70.16
  written, MiB/s:   46.78

General statistics:
  total time:       30.0056s
  total number of events: 511931

Latency (ms):
  min:              0.02
  avg:              0.06
  max:              34.79
  95th percentile:  0.14
  sum:              29872.29

Threads fairness:
  events (avg/stddev): 511931.0000/0.00
  execution time (avg/stddev): 29.8723/0.00
```

Case 5:

Threads started!

File operations:
 reads/s: 4369.00
 writes/s: 2912.65
 fsyncs/s: 9320.60

Throughput:
 read, MiB/s: 68.27
 written, MiB/s: 45.51

General statistics:
 total time: 30.0061s
 total number of events: 498050

Latency (ms):
 min: 0.02
 avg: 0.06
 max: 5.91
 95th percentile: 0.14
 sum: 29871.97

Threads fairness:
 events (avg/stddev): 498050.0000/0.00
 execution time (avg/stddev): 29.8720/0.00

Scenario 3

Command used:

```
sysbench --test=fileio --file-total-size=4G --file-test-mode=rndrw --max-time=30 --max-requests=0 --file-extra-flags=direct prepare
```

```
sysbench --test=fileio --file-total-size=4G --file-test-mode=rndrw --max-time=30 --max-requests=0 --file-extra-flags=direct run
```

```
sleep 60
```

```
sysbench --test=fileio --file-total-size=4G --file-test-mode=rndrw --max-time=30 --max-requests=0 --file-extra-flags=direct cleanup
```

Case 1:

```
Threads started!

File operations:
  reads/s:          4559.16
  writes/s:         3039.44
  fsyncs/s:         9726.23

Throughput:
  read, MiB/s:      71.24
  written, MiB/s:   47.49

General statistics:
  total time:       30.0051s
  total number of events: 519713

Latency (ms):
  min:              0.02
  avg:              0.06
  max:              29.01
  95th percentile:  0.14
  sum:              29869.61

Threads fairness:
  events (avg/stddev): 519713.0000/0.00
  execution time (avg/stddev): 29.8696/0.00
```

Case 2:

```
Threads started!

File operations:
  reads/s:          4215.41
  writes/s:         2810.27
  fsyncs/s:         8996.71

Throughput:
  read, MiB/s:      65.87
  written, MiB/s:   43.91

General statistics:
  total time:       30.0051s
  total number of events: 480634

Latency (ms):
  min:              0.02
  avg:              0.06
  max:              44.86
  95th percentile:  0.16
  sum:              29880.58

Threads fairness:
  events (avg/stddev): 480634.0000/0.00
  execution time (avg/stddev): 29.8806/0.00
```

Case 3:

Threads started!

File operations:

reads/s:	4395.94
writes/s:	2930.65
fsyncs/s:	9380.63

Throughput:

read, MiB/s:	68.69
written, MiB/s:	45.79

General statistics:

total time:	30.0052s
total number of events:	501183

Latency (ms):

min:	0.02
avg:	0.06
max:	2.21
95th percentile:	0.14
sum:	29873.69

Threads fairness:

events (avg/stddev):	501183.0000/0.00
execution time (avg/stddev):	29.8737/0.00

Case 4:

```
Threads started!

File operations:
  reads/s:          4469.33
  writes/s:         2979.57
  fsyncs/s:         9538.56

Throughput:
  read, MiB/s:      69.83
  written, MiB/s:   46.56

General statistics:
  total time:       30.0049s
  total number of events: 509587

Latency (ms):
  min:              0.02
  avg:              0.06
  max:              4.33
  95th percentile:  0.14
  sum:              29868.81

Threads fairness:
  events (avg/stddev): 509587.0000/0.00
  execution time (avg/stddev): 29.8688/0.00
```

Case 5:

```
Threads started!

File operations:
  reads/s:          4497.25
  writes/s:         2998.16
  fsyncs/s:         9598.36

Throughput:
  read, MiB/s:      70.27
  written, MiB/s:   46.85

General statistics:
  total time:       30.0045s
  total number of events: 512771

Latency (ms):
  min:              0.02
  avg:              0.06
  max:              10.40
  95th percentile:  0.14
  sum:              29870.55

Threads fairness:
  events (avg/stddev): 512771.0000/0.00
  execution time (avg/stddev): 29.8706/0.00
```

DOCKER Testing:

- CPU Testing

Scenario 1

The CPU is tested using the cpu-max-prime where the max prime number is found under the given limit.

Command used: CPU Time - sysbench --test=cpu --cpu-max-prime=10000 run

Case 1:

```
Running the test with following options:  
Number of threads: 1  
Initializing random number generator from current time  
  
Prime numbers limit: 10000  
  
Initializing worker threads...  
  
Threads started!  
  
CPU speed:  
    events per second: 8528.70  
  
General statistics:  
    total time:                      10.0002s  
    total number of events:           85295  
  
Latency (ms):  
    min:                            0.11  
    avg:                            0.12  
    max:                            1.15  
    95th percentile:                0.12  
    sum:                            9977.54  
  
Threads fairness:  
    events (avg/stddev):            85295.0000/0.00  
    execution time (avg/stddev):    9.9775/0.00
```

Case 2:

Running the test with following options:
Number of threads: 1
Initializing random number generator from current time

Prime numbers limit: 10000

Initializing worker threads...

Threads started!

CPU speed:
events per second: 8543.34

General statistics:
total time: 10.0002s
total number of events: 85441

Latency (ms):
min: 0.11
avg: 0.12
max: 0.27
95th percentile: 0.12
sum: 9976.81

Threads fairness:
events (avg/stddev): 85441.0000/0.00
execution time (avg/stddev): 9.9768/0.00

Case 3:

Running the test with following options:

Number of threads: 1

Initializing random number generator from current time

Prime numbers limit: 10000

Initializing worker threads...

Threads started!

CPU speed:

events per second: 8505.12

General statistics:

total time: 10.0002s

total number of events: 85058

Latency (ms):

min: 0.11

avg: 0.12

max: 0.63

95th percentile: 0.12

sum: 9976.66

Threads fairness:

events (avg/stddev): 85058.0000/0.00

execution time (avg/stddev): 9.9767/0.00

Case 4:

Running the test with following options:

Number of threads: 1

Initializing random number generator from current time

Prime numbers limit: 10000

Initializing worker threads...

Threads started!

CPU speed:

events per second: 8542.01

General statistics:

total time:	10.0001s
total number of events:	85427

Latency (ms):

min:	0.11
avg:	0.12
max:	0.30
95th percentile:	0.12
sum:	9975.95

Threads fairness:

events (avg/stddev):	85427.0000/0.00
execution time (avg/stddev):	9.9760/0.00

Case 5:

```
Running the test with following options:  
Number of threads: 1  
Initializing random number generator from current time  
  
Prime numbers limit: 10000  
  
Initializing worker threads...  
  
Threads started!  
  
CPU speed:  
    events per second: 8551.12  
  
General statistics:  
    total time:          10.0001s  
    total number of events: 85518  
  
Latency (ms):  
    min:                 0.11  
    avg:                 0.12  
    max:                 0.62  
    95th percentile:    0.12  
    sum:                9980.37  
  
Threads fairness:  
    events (avg/stddev): 85518.0000/0.00  
    execution time (avg/stddev): 9.9804/0.00
```

Scenario 2

The CPU is tested using the cpu-max-prime where the max prime number is found under the given limit.
Command used: CPU Time - sysbench --test=cpu --cpu-max-prime=30000 run

Case 1:

Running the test with following options:
Number of threads: 1
Initializing random number generator from current time

Prime numbers limit: 30000

Initializing worker threads...

Threads started!

CPU speed:
events per second: 2106.10

General statistics:
total time: 10.0005s
total number of events: 21064

Latency (ms):
min: 0.46
avg: 0.47
max: 3.13
95th percentile: 0.49
sum: 9983.21

Threads fairness:
events (avg/stddev): 21064.0000/0.00
execution time (avg/stddev): 9.9832/0.00

Case 2:

Running the test with following options:
Number of threads: 1
Initializing random number generator from current time

Prime numbers limit: 30000

Initializing worker threads...

Threads started!

CPU speed:
events per second: 2114.71

General statistics:
total time: 10.0003s
total number of events: 21149

Latency (ms):
min: 0.46
avg: 0.47
max: 1.33
95th percentile: 0.49
sum: 9984.78

Threads fairness:
events (avg/stddev): 21149.0000/0.00
execution time (avg/stddev): 9.9848/0.00

Case 3:

```
Running the test with following options:  
Number of threads: 1  
Initializing random number generator from current time  
  
Prime numbers limit: 30000  
  
Initializing worker threads...  
  
Threads started!  
  
CPU speed:  
    events per second: 2092.39  
  
General statistics:  
    total time:          10.0003s  
    total number of events: 20926  
  
Latency (ms):  
    min:                0.46  
    avg:                0.48  
    max:                2.00  
    95th percentile:    0.51  
    sum:                9987.05  
  
Threads fairness:  
    events (avg/stddev): 20926.0000/0.00  
    execution time (avg/stddev): 9.9871/0.00
```

Case 4:

Running the test with following options:

Number of threads: 1

Initializing random number generator from current time

Prime numbers limit: 30000

Initializing worker threads...

Threads started!

CPU speed:

events per second: 2115.52

General statistics:

total time: 10.0003s

total number of events: 21158

Latency (ms):

min: 0.46

avg: 0.47

max: 0.91

95th percentile: 0.49

sum: 9985.27

Threads fairness:

events (avg/stddev): 21158.0000/0.00

execution time (avg/stddev): 9.9853/0.00

Case 5:

```
Running the test with following options:  
Number of threads: 1  
Initializing random number generator from current time  
  
Prime numbers limit: 30000  
  
Initializing worker threads...  
  
Threads started!  
  
CPU speed:  
    events per second: 2116.59  
  
General statistics:  
    total time:          10.0004s  
    total number of events: 21168  
  
Latency (ms):  
    min:                 0.46  
    avg:                 0.47  
    max:                 0.81  
    95th percentile:     0.49  
    sum:                9985.10  
  
Threads fairness:  
    events (avg/stddev): 21168.0000/0.00  
    execution time (avg/stddev): 9.9851/0.00
```

Scenario 3

The CPU is tested using the cpu-max-prime where the max prime number is found under the given limit.
Command used: CPU Time - sysbench --test=cpu --cpu-max-prime=50000 run

Case 1:

Running the test with following options:
Number of threads: 1
Initializing random number generator from current time

Prime numbers limit: 50000

Initializing worker threads...

Threads started!

CPU speed:
events per second: 1089.28

General statistics:
total time: 10.0010s
total number of events: 10895

Latency (ms):
min: 0.89
avg: 0.92
max: 4.22
95th percentile: 0.97
sum: 9989.21

Threads fairness:
events (avg/stddev): 10895.0000/0.00
execution time (avg/stddev): 9.9892/0.00

Case 2:

```
Running the test with following options:  
Number of threads: 1  
Initializing random number generator from current time  
  
Prime numbers limit: 50000  
  
Initializing worker threads...  
  
Threads started!  
  
CPU speed:  
    events per second: 1097.86  
  
General statistics:  
    total time: 10.0003s  
    total number of events: 10980  
  
Latency (ms):  
    min: 0.89  
    avg: 0.91  
    max: 1.74  
    95th percentile: 0.94  
    sum: 9987.72  
  
Threads fairness:  
    events (avg/stddev): 10980.0000/0.00  
    execution time (avg/stddev): 9.9877/0.00
```

Case 3:

Running the test with following options:
Number of threads: 1
Initializing random number generator from current time

Prime numbers limit: 50000

Initializing worker threads...

Threads started!

CPU speed:
events per second: 1100.34

General statistics:
total time: 10.0009s
total number of events: 11005

Latency (ms):
min: 0.89
avg: 0.91
max: 1.32
95th percentile: 0.94
sum: 9989.68

Threads fairness:
events (avg/stddev): 11005.0000/0.00
execution time (avg/stddev): 9.9897/0.00

Case 4:

Running the test with following options:
Number of threads: 1
Initializing random number generator from current time

Prime numbers limit: 50000

Initializing worker threads...

Threads started!

CPU speed:
events per second: 1088.13

General statistics:
total time: 10.0009s
total number of events: 10883

Latency (ms):
min: 0.89
avg: 0.92
max: 3.76
95th percentile: 0.97
sum: 9988.94

Threads fairness:
events (avg/stddev): 10883.0000/0.00
execution time (avg/stddev): 9.9889/0.00

Case 5:

```
Running the test with following options:
Number of threads: 1
Initializing random number generator from current time

Prime numbers limit: 50000

Initializing worker threads...

Threads started!

CPU speed:
events per second: 1094.34

General statistics:
total time: 10.0007s
total number of events: 10945

Latency (ms):
min: 0.89
avg: 0.91
max: 1.62
95th percentile: 0.97
sum: 9987.60

Threads fairness:
events (avg/stddev): 10945.0000/0.00
execution time (avg/stddev): 9.9876/0.00

• File I/O Testing
```

Scenario 1

Command used:

```
sysbench --test=fileio --file-total-size=2G --file-test-mode=rndrw --max-time=30 --max-requests=0 --file-extra-flags=direct prepare
sysbench --test=fileio --file-total-size=2G --file-test-mode=rndrw --max-time=30 --max-requests=0 --
file-extra-flags=direct run
sleep 60
sysbench --test=fileio --file-total-size=2G --file-test-mode=rndrw --max-time=30 --max-requests=0 --
file-extra-flags=direct cleanup
```

Case 1:

Initializing worker threads...

Threads started!

File operations:

reads/s:	3824.70
writes/s:	2549.78
fsyncs/s:	8160.15

Throughput:

read, MiB/s:	59.76
written, MiB/s:	39.84

General statistics:

total time:	30.0066s
total number of events:	436017

Latency (ms):

min:	0.03
avg:	0.07
max:	7.51
95th percentile:	0.15
sum:	29824.06

Threads fairness:

events (avg/stddev):	436017.0000/0.00
execution time (avg/stddev):	29.8241/0.00

Case 2:

Initializing worker threads...

Threads started!

File operations:

reads/s:	3598.05
writes/s:	2398.72
fsyncs/s:	7678.47

Throughput:

read, MiB/s:	56.22
written, MiB/s:	37.48

General statistics:

total time:	30.0053s
total number of events:	410211

Latency (ms):

min:	0.03
avg:	0.07
max:	5.82
95th percentile:	0.16
sum:	29839.99

Threads fairness:

events (avg/stddev):	410211.0000/0.00
execution time (avg/stddev):	29.8400/0.00

Case 3:

Initializing worker threads...

Threads started!

File operations:

reads/s:	3507.27
writes/s:	2338.18
fsyncs/s:	7485.69

Throughput:

read, MiB/s:	54.80
written, MiB/s:	36.53

General statistics:

total time:	30.0055s
total number of events:	399889

Latency (ms):

min:	0.03
avg:	0.07
max:	12.45
95th percentile:	0.16
sum:	29848.65

Threads fairness:

events (avg/stddev):	399889.0000/0.00
execution time (avg/stddev):	29.8486/0.00

Case 4:

Initializing worker threads...

Threads started!

File operations:

reads/s:	3517.38
writes/s:	2344.92
fsyncs/s:	7506.65

Throughput:

read, MiB/s:	54.96
written, MiB/s:	36.64

General statistics:

total time:	30.0047s
total number of events:	401011

Latency (ms):

min:	0.03
avg:	0.07
max:	7.06
95th percentile:	0.16
sum:	29852.37

Threads fairness:

events (avg/stddev):	401011.0000/0.00
execution time (avg/stddev):	29.8524/0.00

Case 5:

Initializing worker threads...

Threads started!

File operations:

reads/s:	3545.35
writes/s:	2363.56
fsyncs/s:	7566.70

Throughput:

read, MiB/s:	55.40
written, MiB/s:	36.93

General statistics:

total time:	30.0049s
total number of events:	404215

Latency (ms):

min:	0.03
avg:	0.07
max:	7.48
95th percentile:	0.16
sum:	29854.61

Threads fairness:

events (avg/stddev):	404215.0000/0.00
execution time (avg/stddev):	29.8546/0.00

Scenario 2

Command used:

```
sysbench --test=fileio --file-total-size=3G --file-test-mode=rndrw --max-time=30 --max-requests=0 --file-extra-flags=direct prepare
```

```
sysbench --test=fileio --file-total-size=3G --file-test-mode=rndrw --max-time=30 --max-requests=0 --file-extra-flags=direct run
```

```
sleep 60
```

```
sysbench --test=fileio --file-total-size=3G --file-test-mode=rndrw --max-time=30 --max-requests=0 --file-extra-flags=direct cleanup
```

Case 1:

Initializing worker threads...

Threads started!

File operations:

reads/s:	3644.10
writes/s:	2429.40
fsyncs/s:	7776.42

Throughput:

read, MiB/s:	56.94
written, MiB/s:	37.96

General statistics:

total time:	30.0060s
total number of events:	415461

Latency (ms):

min:	0.03
avg:	0.07
max:	26.26
95th percentile:	0.16
sum:	29869.96

Threads fairness:

events (avg/stddev):	415461.0000/0.00
execution time (avg/stddev):	29.8700/0.00

Case 2:

Initializing worker threads...

Threads started!

File operations:

reads/s:	3765.77
writes/s:	2510.53
fsyncs/s:	8036.48

Throughput:

read, MiB/s:	58.84
written, MiB/s:	39.23

General statistics:

total time:	30.0064s
total number of events:	429358

Latency (ms):

min:	0.03
avg:	0.07
max:	5.94
95th percentile:	0.15
sum:	29866.16

Threads fairness:

events (avg/stddev):	429358.0000/0.00
execution time (avg/stddev):	29.8662/0.00

Case 3:

Initializing worker threads...

Threads started!

File operations:

reads/s:	3701.52
writes/s:	2467.69
fsyncs/s:	7900.34

Throughput:

read, MiB/s:	57.84
written, MiB/s:	38.56

General statistics:

total time:	30.0052s
total number of events:	422040

Latency (ms):

min:	0.03
avg:	0.07
max:	20.67
95th percentile:	0.15
sum:	29872.39

Threads fairness:

events (avg/stddev):	422040.0000/0.00
execution time (avg/stddev):	29.8724/0.00

Case 4:

Initializing worker threads...

Threads started!

File operations:

reads/s:	3478.70
writes/s:	2319.12
fsyncs/s:	7422.15

Throughput:

read, MiB/s:	54.35
written, MiB/s:	36.24

General statistics:

total time:	30.0069s
total number of events:	396570

Latency (ms):

min:	0.03
avg:	0.08
max:	27.58
95th percentile:	0.16
sum:	29851.72

Threads fairness:

events (avg/stddev):	396570.0000/0.00
execution time (avg/stddev):	29.8517/0.00

Case 5:

Initializing worker threads...

Threads started!

File operations:

reads/s:	3488.89
writes/s:	2325.92
fsyncs/s:	7443.68

Throughput:

read, MiB/s:	54.51
written, MiB/s:	36.34

General statistics:

total time:	30.0061s
total number of events:	397715

Latency (ms):

min:	0.03
avg:	0.08
max:	4.69
95th percentile:	0.16
sum:	29857.11

Threads fairness:

events (avg/stddev):	397715.0000/0.00
execution time (avg/stddev):	29.8571/0.00

Scenario 3

Command used:

```
sysbench --test=fileio --file-total-size=4G --file-test-mode=rndrw --max-time=30 --max-requests=0 --file-extra-flags=direct prepare
```

```
sysbench --test=fileio --file-total-size=4G --file-test-mode=rndrw --max-time=30 --max-requests=0 --file-extra-flags=direct run
```

```
sleep 60
```

```
sysbench --test=fileio --file-total-size=4G --file-test-mode=rndrw --max-time=30 --max-requests=0 --file-extra-flags=direct cleanup
```

Case 1:

Initializing worker threads...

Threads started!

File operations:

reads/s:	3495.32
writes/s:	2330.21
fsyncs/s:	7457.05

Throughput:

read, MiB/s:	54.61
written, MiB/s:	36.41

General statistics:

total time:	30.0053s
total number of events:	398427

Latency (ms):

min:	0.03
avg:	0.07
max:	8.64
95th percentile:	0.16
sum:	29849.73

Threads fairness:

events (avg/stddev):	398427.0000/0.00
execution time (avg/stddev):	29.8497/0.00

Case 2:

Initializing worker threads...

Threads started!

File operations:

reads/s:	3525.96
writes/s:	2350.64
fsyncs/s:	7524.61

Throughput:

read, MiB/s:	55.09
written, MiB/s:	36.73

General statistics:

total time:	30.0066s
total number of events:	402004

Latency (ms):

min:	0.03
avg:	0.07
max:	6.64
95th percentile:	0.16
sum:	29860.79

Threads fairness:

events (avg/stddev):	402004.0000/0.00
execution time (avg/stddev):	29.8608/0.00

Case 3:

Initializing worker threads...

Threads started!

File operations:

reads/s:	3463.64
writes/s:	2309.11
fsyncs/s:	7392.57

Throughput:

read, MiB/s:	54.12
written, MiB/s:	36.08

General statistics:

total time:	30.0057s
total number of events:	394915

Latency (ms):

min:	0.03
avg:	0.08
max:	50.32
95th percentile:	0.17
sum:	29856.40

Threads fairness:

events (avg/stddev):	394915.0000/0.00
execution time (avg/stddev):	29.8564/0.00

Case 4:

```
Using random dev seed  
Initializing worker threads...
```

Threads started!

File operations:

reads/s:	3501.37
writes/s:	2334.24
fsyncs/s:	7470.81

Throughput:

read, MiB/s:	54.71
written, MiB/s:	36.47

General statistics:

total time:	30.0049s
total number of events:	399137

Latency (ms):

min:	0.03
avg:	0.07
max:	25.18
95th percentile:	0.16
sum:	29851.33

Threads fairness:

events (avg/stddev):	399137.0000/0.00
execution time (avg/stddev):	29.8513/0.00

Case 5:

Initializing worker threads...

Threads started!

File operations:

reads/s:	3455.20
writes/s:	2303.46
fsyncs/s:	7371.08

Throughput:

read, MiB/s:	53.99
written, MiB/s:	35.99

General statistics:

total time:	30.0064s
total number of events:	393856

Latency (ms):

min:	0.03
avg:	0.08
max:	26.89
95th percentile:	0.16
sum:	29859.52

Threads fairness:

events (avg/stddev):	393856.0000/0.00
execution time (avg/stddev):	29.8595/0.00

OS and System Virtualization performances result and comparisons

CPU Performance Results

QEMU Ubuntu Virtual Machine

- Scenario 1 - --cpu-max-prime=10000

Case	Total Time	CPU Speed	Avg. Latency
1	10.0002	11027.52	0.09
2	10.0001	11067.66	0.09
3	10.0002	11059.83	0.09
4	10.0002	11048.16	0.09
5	10.0002	11058.12	0.09
Minimum	10.0001	11027.52	0.09
Maximum	10.0002	11067.66	0.09
Avgverage	10.00018	11052.258	0.09
StdDev	4.47214E-05	15.47321945	0

- Scenario 2 – --cpu-max-prime=30000

Case	Total Time	CPU Speed	Avg. Latency
1	10.0003	2478.65	0.4
2	10.0002	2485.7	0.4
3	10.0005	2482.32	0.4
4	10.0002	2487.49	0.4
5	10.0005	2486.53	0.4
Minimum	10.0002	2478.65	0.4
Maximum	10.0005	2487.49	0.4
Avgverage	10.00034	2484.138	0.4
StdDev	0.000151658	3.633891853	0

- Scenario 3 – --cpu-max-prime=50000

Case	Total Time	CPU Speed	Avg. Latency
1	10.0005	1232.77	0.81

2	10.0003	1235.48	0.81
3	10.0007	1235.73	0.81
4	10.0005	1235.96	0.81
5	10.0003	1234.19	0.81
Minimum	10.0003	1232.77	0.81
Maximum	10.0007	1235.96	0.81
Average	10.00046	1234.826	0.81
StdDev	0.000167332	1.338144237	1.2413E-16

File I/O Performance results

- Scenario 1 – -file-size=2G

Case	Read Throughput (MiB/s)	Write Throughput (MiB/s)	Total Time (s)
1	78.8	52.53	30.0047
2	62.56	41.71	30.0056
3	68.42	45.61	30.0058
4	67.15	44.77	30.0052
5	67.36	44.91	30.0048
Minimum	62.56	41.71	30.0047
Maximum	78.8	52.53	30.0058
Average	68.858	45.906	30.00522
StdDev	5.997109304	3.995432392	0.00048166

- Scenario 2 – -file-size=3G

Case	Read Throughput (MiB/s)	Write Throughput (MiB/s)	Total Time (s)
1	70.21	46.81	30.0052
2	65.66	43.77	30.0053
3	67.01	44.68	30.0055
4	70.16	46.78	30.0056
5	68.27	45.51	30.0061
Minimum	65.66	43.77	30.0052
Maximum	70.21	46.81	30.0061
Average	68.262	45.51	30.00554
StdDev	1.983373389	1.324707515	0.000350714

- Scenario 3 – -file-size=4G

Case	Read Throughput (MiB/s)	Write Throughput (MiB/s)	Total Time (s)
1	71.24	47.49	30.0051
2	65.87	43.91	30.0051
3	68.69	45.79	30.0052
4	69.83	46.56	30.0049
5	70.27	46.85	30.0045
Minimum	65.87	43.91	30.0045
Maximum	71.24	47.49	30.0052
Avgverage	69.18	46.12	30.00496
StdDev	2.0646065	1.378078372	0.000279285

DOCKER

CPU Performance Results

- Scenario 1 – --cpu-max-prime=10000

Case	Total Time (s)	CPU Speed (events/sec)	Avg. Latency (ms)
1	10.0002	8528.7	0.12
2	10.0002	8543.34	0.12
3	10.0002	8505.12	0.12
4	10.0001	8542.01	0.12
5	10.0001	8551.12	0.12
Minimum	10.0001	8505.12	0.12
Maximum	10.0002	8551.12	0.12
Avgverage	10.00016	8534.058	0.12
StdDev	5.47723E-05	18.07358349	0

- Scenario 2 – --cpu-max-prime=30000

Case	Total Time (s)	CPU Speed (events/sec)	Avg. Latency (ms)

1	10.0005	2106.1	0.47
2	10.0003	2114.71	0.47
3	10.0003	2092.39	0.47
4	10.0003	2115.52	0.47
5	10.0004	2116.59	0.47
Minimum	10.0003	2092.39	0.47
Maximum	10.0005	2116.59	0.47
Avgverage	10.00036	2109.062	0.47
StdDev	8.94427E-05	10.21035602	6.20634E-17

- Scenario 3 – --cpu-max-prime=50000

Case	Total Time (s)	CPU Speed (events/sec)	Avg. Latency (ms)
1	10.001	1089.28	0.92
2	10.0003	1097.86	0.92
3	10.0009	1100.34	0.92
4	10.0009	1088.13	0.92
5	10.0007	1094.34	0.92
Minimum	10.0003	1088.13	0.92
Maximum	10.001	1100.34	0.92
Avgverage	10.00076	1093.99	0.92
StdDev	0.000279285	5.290217387	1.24127E-16

File I/O Performance Results

- Scenario 1 – -file-size=2G

Case	Read Throughput (MiB/s)	Write Throughput (MiB/s)	Total Time (s)
1	59.76	39.84	30.0066
2	56.22	37.48	30.0053
3	54.8	36.53	30.0055
4	54.96	36.64	30.0047
5	55.4	36.93	30.0049
Minimum	54.8	36.53	30.0047
Maximum	59.76	39.84	30.0066
Avgverage	56.228	37.484	30.0054
StdDev	2.049858532	1.367490402	0.00074162

- Scenario 2 – -file-size=3G

Case	Read Throughput (MiB/s)	Write Throughput (MiB/s)	Total Time (s)
1	56.94	37.96	30.006
2	58.84	39.23	30.0064
3	57.84	38.56	30.0052
4	54.35	36.23	30.0069
5	54.51	36.34	30.0061
Minimum	54.35	36.23	30.0052
Maximum	58.84	39.23	30.0069
Avgverage	56.496	37.664	30.00612
StdDev	2.002955317	1.33717239	0.000622093

- Scenario 3 – file-size=4G

Case	Read Throughput (MiB/s)	Write Throughput (MiB/s)	Total Time (s)
1	54.61	36.41	30.0053
2	55.09	36.73	30.0066
3	54.12	36.08	30.0057
4	54.71	36.47	30.0049
5	53.99	35.99	30.0064
Minimum	53.99	35.99	30.0049
Maximum	55.09	36.73	30.0066
Avgverage	54.504	36.336	30.00578
StdDev	0.449644304	0.301628911	0.000719027

Performance of Host Machine

The performance of host machine was observed by running the top command in host terminal when the scenarios were running on the VM and Docker.

The % of CPU utilization on host machine while running performance on VM:

The screenshot shows two terminal windows side-by-side. The left window displays system monitoring output from 'top' with the command 'ashokkumarrangababu -- top -- 96x29'. It lists processes like 'gemu-system-' and 'WindowsServer' along with their CPU usage, memory, and state. The right window shows a performance test with the command 'ashokkumarrangababu -- qemu-system-arch64 -accel hvf -cpu cortex-a57 -M virt,highmem=off -m 2G...'. It includes output for 'Prime numbers limit: 50000', 'Threads started!', and 'CPU speed: events per second: 1215.88'. Both windows have tabs for 'zsh', 'exec -it ubuntu bash', 'use -usb -nographic', and 'zsh'.

The % of CPU Utilization on host machine while running performance on Docker

The screenshot shows a single terminal window with the command 'ashokkumarrangababu -- root@73d573809f8d: / com.docker.cli + docker exec -it ubuntu bash - 109x41'. The output is identical to the one in the previous screenshot, showing system monitoring ('top') and a performance test ('qemu-system-arch64'). The window has tabs for 'zsh', 'exec -it ubuntu bash', 'use -usb -nographic', and 'zsh'.

We observe a 100% CPU utilization on host machine while running performance scenarios in both VM and Docker.

Concluding remarks based on various scenarios:

Based on the various test scenarios on my machine I can conclude that the System virtualization is better than OS Virtualization as we observe that docker outperforms on a slight higher rate than the QEMU in all three scenarios in CPU performance testing and File I/O testing.

Github Repo Info:

Account Name: APadmakshi

Repo-Name: COEN-241---Fall-2022

Folder: HW1

Link to Repo: <https://github.com/APadmakshi/COEN-241---Fall-2022>