

Fio: Flexible I/O Tester

Vincent Fu
Samsung Semiconductor

Northern Virginia Linux Users Group
April 2024

Who Am I?


- First computer was a TI-99/4A
- Learned BASIC, Pascal, COBOL, C, x86 assembly as a kid
- 15-year detour as an academic social scientist
- Started career at Fusion-io (early PCIe-attached SSDs), now at Samsung Semiconductor's Global Open ecoSystem Team



Fio: Background

- First commit Dec 2005 by Jens Axboe
- <https://github.com/axboe/fio>
 - 4.8K stars on GitHub, 1.2K forks, ~2000 gits clones a day
- Used by storage vendors and in projects like `blktests` and `fstests`
- From the README:
 - Fio was originally written to save me the hassle of writing special test case programs when I wanted to test a specific workload, either for performance reasons or to find/reproduce a bug.... I needed a tool that would be able to simulate a given I/O workload without resorting to writing a tailored test case again and again.

Fio: My Involvement

- At Fusion-io we used fio for performance testing The Fusion-io logo consists of a stylized, multi-pointed star or flower-like symbol to the left of the text "FUSION-io". The text is in a bold, sans-serif font, with "FUSION" in all caps and "-io" in lowercase.
- Eventually I started helping coworkers with fio features/bug fixes
- Co-maintainer since 2022
- Fio blog:
<https://github.com/vincentkfu/fio-blog/wiki>

Fio Basics: Availability

- Available from your distribution's package manager

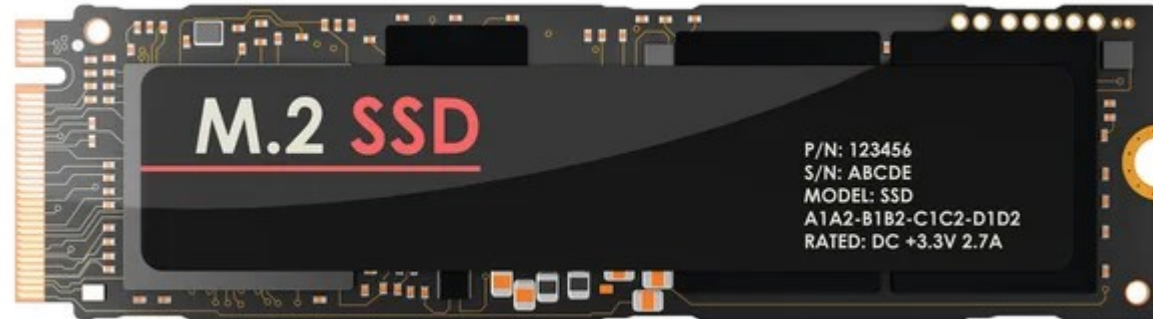
```
$ sudo dnf install fio
```

```
$ sudo apt install fio
```

- Can of course build it yourself
- Documentation: <https://fio.readthedocs.io/>

What can fio do for me?

- Validating a new storage device
 - Data persistence
 - Performance



Data Persistence: Setup

- Full write with verify to make sure device works

- First, identify device to test using `lsblk`

```
root@fedora:~# lsblk
```

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINTS
sr0	11:0	1	1024M	0	rom	
zram0	252:0	0	8G	0	disk	[SWAP]
nvme1n1	259:0	0	110.3G	0	disk	
nvme2n1	259:1	0	119.2G	0	disk	
nvme0n1	259:2	0	119.2G	0	disk	
├─nvme0n1p1	259:3	0	600M	0	part	/boot/efi
├─nvme0n1p2	259:4	0	1G	0	part	/boot
├─nvme0n1p3	259:5	0	117.7G	0	part	
└─fedora_fedora-root	253:0	0	117.7G	0	lvm	/

Data Persistence: Job File

```
[test]
```

```
bs=64k
```

```
rw=write
```

```
verify=crc32c
```

```
filename=/dev/nvme1n1
```


Testing A New Device: Job Run

```
root@fedora:~# fio surface-scan.fio
test: (g=0): rw=write, bs=(R) 64.0KiB-64.0KiB, (W) 64.0KiB-64.0KiB, (T) 64.0KiB-64.0KiB, ioengine=psync, iodepth=1
fio-3.35
Starting 1 process
Jobs: 1 (f=1): [f(1)][100.0%][eta 00m:00s]
test: (groupid=0, jobs=1): err= 0: pid=2194: Mon Mar 25 16:07:52 2024
read: IOPS=24.4k, BW=1522MiB/s (1596MB/s)(110GiB/74177msec)
  clat (usec): min=4, max=3271, avg=30.00, stdev=47.41
    lat (usec): min=4, max=3271, avg=30.04, stdev=47.41
  clat percentiles (usec):
  | 1.00th=[ 6], 5.00th=[ 6], 10.00th=[ 6], 20.00th=[ 6],
  | 30.00th=[ 6], 40.00th=[ 6], 50.00th=[ 6], 60.00th=[ 7],
  | 70.00th=[ 7], 80.00th=[ 85], 90.00th=[ 96], 95.00th=[ 104],
  | 99.00th=[ 281], 99.50th=[ 289], 99.90th=[ 289], 99.95th=[ 289],
  | 99.99th=[ 293]
write: IOPS=13.2k, BW=824MiB/s (864MB/s)(110GiB/137063msec); 0 zone resets
  clat (usec): min=19, max=30494, avg=55.05, stdev=451.39
    lat (usec): min=36, max=30535, avg=75.42, stdev=451.53
  clat percentiles (usec):
  | 1.00th=[ 23], 5.00th=[ 24], 10.00th=[ 24], 20.00th=[ 26],
  | 30.00th=[ 27], 40.00th=[ 29], 50.00th=[ 31], 60.00th=[ 34],
  | 70.00th=[ 36], 80.00th=[ 42], 90.00th=[ 50], 95.00th=[ 56],
  | 99.00th=[ 92], 99.50th=[ 117], 99.90th=[ 8029], 99.95th=[ 8979],
  | 99.99th=[16188]
bw ( K1B/s): min=204544, max=1529344, per=100.00%, avg=843908.91, stdev=192888.85, samples=274
iops      : min= 3196, max=23896, avg=13186.00, stdev=3013.90, samples=274
lat (usec)  : 10=37.29%, 20=0.21%, 50=45.22%, 100=13.13%, 250=3.44%
lat (usec)  : 500=0.59%, 750=0.01%
lat (msec)  : 2=0.01%, 4=0.01%, 10=0.11%, 20=0.01%, 50=0.01%
cpu         : usr=26.12%, sys=44.91%, ctx=373039, majf=1, minf=49420
IO depths   : 1=100.0%, 2=0.0%, 4=0.0%, 8=0.0%, 16=0.0%, 32=0.0%, >=64=0.0%
submit     : 0=0.0%, 4=100.0%, 8=0.0%, 16=0.0%, 32=0.0%, 64=0.0%, >=64=0.0%
complete   : 0=0.0%, 4=100.0%, 8=0.0%, 16=0.0%, 32=0.0%, 64=0.0%, >=64=0.0%
issued rwts: total=1806800,1806800,0,0 short=0,0,0,0 dropped=0,0,0,0
latency    : target=0, window=0, percentile=100.00%, depth=1

Run status group 0 (all jobs):
  READ: bw=1522MiB/s (1596MB/s), 1522MiB/s-1522MiB/s (1596MB/s-1596MB/s), io=110GiB (118GB), run=74177-74177msec
  WRITE: bw=824MiB/s (864MB/s), 824MiB/s-824MiB/s (864MB/s-864MB/s), io=110GiB (118GB), run=137063-137063msec

Disk stats (read/write):
nvme1n1: ios=903078/917106, merge=0/27991694, ticks=141508/71224856, in_queue=71366364, util=97.79%
```

Performance Testing

- Asynchronous I/O with high iodepth
- O_DIRECT
- Multiple jobs
- ramp_time

Performance Testing: Job File

<pre>[test] ioengine=io_uring filename=/dev/nvme1n1 numjobs=4 group_reporting=1 runtime=30s ramp_time=5s</pre>	<pre>direct=1 bs=4k rw=randread iodepth=64 time_based</pre>
--	---

Performance Testing: Job Run

```
vincent@fedora:~/fio-dev/novalug$ sudo fio perf.fio
test: (g=0): rw=randread, bs=(R) 4096B-4096B, (W) 4096B-4096B, (T) 4096B-4096B, ioengine=io_uring, iodepth=64
...
fio-3.35
Starting 4 processes
Jobs: 4 (f=4): [r(4)][100.0%][r=1679MiB/s][r=430k IOPS][eta 00m:00s]
test: (groupid=0, jobs=4): err= 0: pid=19583: Fri Apr 12 12:51:58 2024
  read: IOPS=429k, BW=1675MiB/s (1756MB/s)(49.1GiB/30001msec)
    slat (nsec): min=1623, max=12041k, avg=2518.20, stdev=10755.74
    clat (usec): min=13, max=21216, avg=593.84, stdev=278.77
      lat (usec): min=15, max=21218, avg=596.36, stdev=279.03
    clat percentiles (usec):
      | 1.00th=[ 239], 5.00th=[ 412], 10.00th=[ 437], 20.00th=[ 441],
      | 30.00th=[ 441], 40.00th=[ 445], 50.00th=[ 449], 60.00th=[ 461],
      | 70.00th=[ 824], 80.00th=[ 881], 90.00th=[ 898], 95.00th=[ 906],
      | 99.00th=[ 1037], 99.50th=[ 1532], 99.90th=[ 3195], 99.95th=[ 3589],
      | 99.99th=[ 5473]
  bw ( MiB/s): min= 1127, max= 2250, per=100.00%, avg=1676.81, stdev=85.42, samples=236
  iops        : min=288680, max=576012, avg=429263.17, stdev=21866.80, samples=236
  lat (usec)  : 20=0.01%, 50=0.02%, 100=0.07%, 250=1.12%, 500=61.54%
  lat (usec)  : 750=6.28%, 1000=29.76%
  lat (msec)  : 2=0.86%, 4=0.32%, 10=0.03%, 20=0.01%, 50=0.01%
  cpu         : usr=9.88%, sys=26.97%, ctx=6865446, majf=0, minf=145
  IO depths   : 1=0.0%, 2=0.0%, 4=0.0%, 8=0.0%, 16=0.0%, 32=0.0%, >=64=100.0%
    submit    : 0=0.0%, 4=100.0%, 8=0.0%, 16=0.0%, 32=0.0%, 64=0.0%, >=64=0.0%
    complete  : 0=0.0%, 4=100.0%, 8=0.0%, 16=0.0%, 32=0.0%, 64=0.1%, >=64=0.0%
    issued rwts: total=12864980,0,0,0 short=0,0,0,0 dropped=0,0,0,0
    latency   : target=0, window=0, percentile=100.00%, depth=64

Run status group 0 (all jobs):
  READ: bw=1675MiB/s (1756MB/s), 1675MiB/s-1675MiB/s (1756MB/s-1756MB/s), io=49.1GiB (52.7GB), run=30001-30001msec

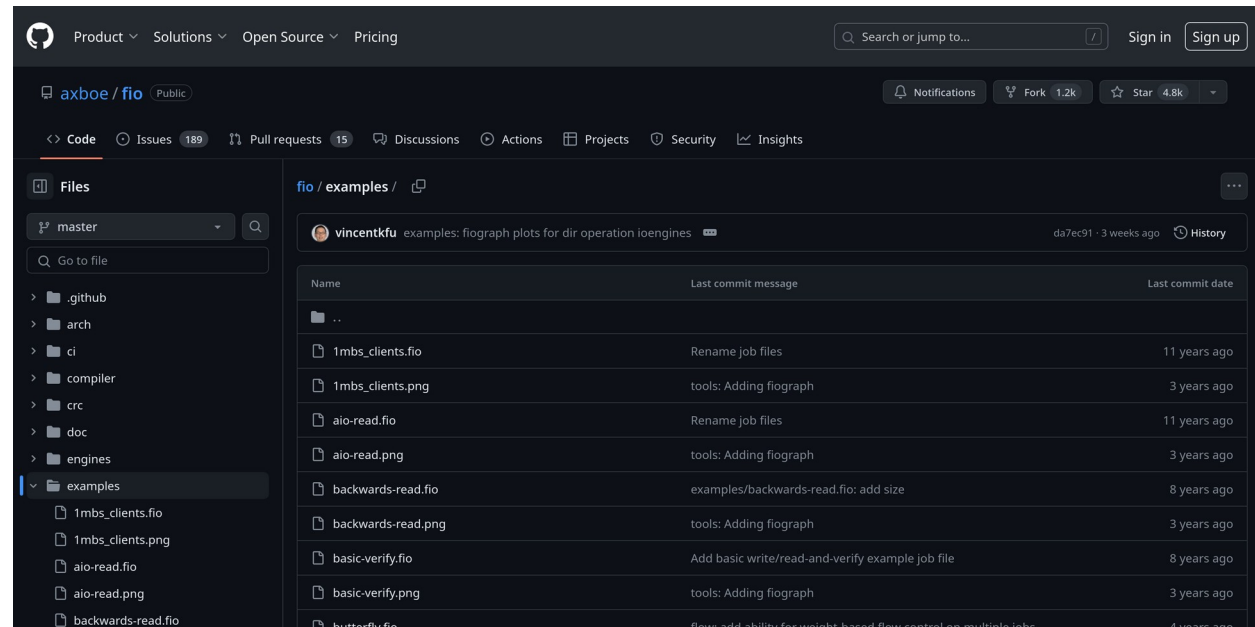
Disk stats (read/write):
  nvme1n1: ios=14907042/0, merge=0/0, ticks=8485250/0, in_queue=8485250, util=99.82%
```

Fio basics: running a job

- Job name
- `ioengine`: interface for I/O requests
- How much I/O to do?
 - `time_based` or `capacity based`
- Data direction: `[rand]{read,write,trim}`

Noteworthy Features

- Specify job on command line
- Logging: IOPS, bw, latency
- Steady state detection
- JSON output for easy parsing
- Example job files



Relatively Recent News

- `io_uring_cmd` ioengine
- End-to-end data protection
- FDP support
- Bug fix for performance on AMD systems: [link](#)
- Bug fix for verify on big endian platforms: [link](#)

Getting Involved

- GitHub: <https://github.com/axboe/fio>
- Mailing list: <https://lore.kernel.org/fio/>

```
fio.vger.kernel.org archive mirror
 search help / color / mirror / Atom feed

help understanding the output of fio
2024-04-06 7:03 UTC (13+ messages)

Recent changes (master)
2024-04-03 12:00 UTC

Recent changes (master)
2024-03-27 12:00 UTC

Recent changes (master)
2024-03-26 12:00 UTC

Recent changes (master)
2024-03-23 12:00 UTC

Recent changes (master)
2024-03-22 12:00 UTC
```


Thank you!