

1. Создаем 512-мегабайтный файл, забитый нулями

```
dd if=/dev/zero of=/home/akasiyanik/UNIX/lab1/fakeFS.dat bs=512M count=1
```

```
1+0 records in
1+0 records out
536870912 bytes (537 MB, 512 MiB) copied, 2,45465 s, 219 MB/s
```

2. Создаем в файле с файловую систему ext2

```
mke2fs -t ext2 /home/akasiyanik/UNIX/lab1/fakeFS.dat
```

```
mke2fs 1.42.13 (17-May-2015) Discarding device blocks: done
Creating filesystem with 131072 4k blocks and 32768 inodes
Filesystem UUID: 7f6f1778-4b4a-4433-9c7a-a62a78dd9c05
Superblock backups stored on blocks:
    32768, 98304
```

```
Allocating group tables: done
Writing inode tables: done
Writing superblocks and filesystem accounting information: done
```

3. Создаем mount point

```
sudo mkdir /media/fake
```

4. Монтируем файловую систему к /media/fake

```
sudo mount -t ext2 -o loop ~/UNIX/lab1/fakeFS.dat /media/fake
```

5. Создаем произвольную структуру файлов и папок

```
cd /media/fake
sudo -i
mkdir lab1
mkdir lab2
cd lab1
echo "Hello World" >> hello.txt
```

UPD.:

Ниже представлены результаты выполнения команды dd с параметрами *bs=512M count=1* и *bs=1M count=512*:

```
# time ./test1.sh
bs=512M count=1
1+0 records in
1+0 records out
536870912 bytes (537 MB, 512 MiB) copied, 1,08285 s, 496 MB/s
real 0m1.280s
user 0m0.000s
```

```
sys    0m0.744s
```

```
# time ./test2.sh
```

```
bs=1M count=512
```

```
512+0 records in
```

```
512+0 records out
```

```
536870912 bytes (537 MB, 512 MiB) copied, 0,938339 s, 572 MB/s
```

```
real    0m1.127s
```

```
user    0m0.004s
```

```
sys     0m0.592s
```

Как видим, время выполнения отличается на ~0.1 секунды в пользу *bs=1M count=512*, и, на мой взгляд, данное преимущество в условиях задания не играет сколь-нибудь значимой роли.

Однако, при копировании большого объема данных выбор оптимального размер блока (как правило небольшого по размеру) существенно уменьшит время выполнения операции.

Экспериментальным путем было выявлено, что оптимальным размером блока для моей виртуальной машины можно считать 4096 bytes ([ссылка на скрипт](#)):

```
root@akasiyanik-vb:/home/akasiyanik/UNIX/lab1# ./adv-test.sh test1.dat
```

```
block size : transfer rate
```

```
512 : 283 MB/s
```

```
1024 : 175 MB/s
```

```
2048 : 566 MB/s
```

```
4096 : 2 GB/s
```

```
8192 : 883 MB/s
```

```
16384 : 461 MB/s
```

```
32768 : 672 MB/s
```

```
65536 : 775 MB/s
```

```
131072 : 742 MB/s
```

```
262144 : 532 MB/s
```

```
524288 : 409 MB/s
```

```
1048576 : 564 MB/s
```

```
2097152 : 637 MB/s
```

```
4194304 : 516 MB/s
```

```
8388608 : 637 MB/s
```

```
16777216 : 529 MB/s
```

```
33554432 : 471 MB/s
```

```
67108864 : 589 MB/s
```

Это можно объяснить тем, что большинство современных жестких дисков имеет размер сектора 4KB (4096 bytes) ([ссылка](#)).

P.S. Однако, на практике это не подтвердилось ☺

```
root@akasiyanik-vb:/home/akasiyanik/UNIX/lab1# ./test1.sh
```

```
bs=4096 count=131072
```

```
131072+0 records in
```

```
131072+0 records out
```

```
536870912 bytes (537 MB, 512 MiB) copied, 1,28256 s, 419 MB/s
```