System Administration

Week 03, Segment 2

Filesystems

Department of Computer Science
Stevens Institute of Technology

Jan Schaumann
jschauma@stevens.edu
https://stevens.netmeister.org/615/
Let’s pretend we’re a filesystem…

The naive approach.

Photo by The Lucky Neko on Unsplash
https://unsplash.com/photos/2JcixBiKy3I
ip-10-10-0-13#
Let’s pretend we’re a filesystem…

One cat per bucket, please.
ip-10-10-0-13# printf '😊😊☃️' | dd of=/dev/xbd1 bs=1 seek=8
14+0 records in
14+0 records out
14 bytes transferred in 0.001 secs (14000 bytes/sec)
ip-10-10-0-13# printf "$(dd if=/dev/xbd1 count=1 2>/dev/null)\n"
😊😊😊😊
ip-10-10-0-13# printf "$(dd if=/dev/xbd1 bs=1 count=4 skip=4 2>/dev/null)\n"
😊
ip-10-10-0-13# printf "$(dd if=/dev/xbd1 bs=1 count=4 skip=8 2>/dev/null)\n"
😊

ip-10-10-0-13# printf "$(dd if=/dev/xbd1 bs=1 count=12 skip=8 2>/dev/null)\n"

ip-10-10-0-13# dd if=/dev/xbd1 count=1 2>/dev/null | hexdump -C
00000000  f0 9f 98 b8 f0 9f 98 bb f0 9f 90 88 f0 9f 90 88 | ..................... |
00000010  e2 9d 84 ef b8 8f 00 00 00 00 00 00 00 00 00 00 00 | ..................... |
00000020  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 | ..................... |
*
00000200
ip-10-10-0-13# printf "$(dd if=/dev/xbd1 bs=1 count=13 skip=8 2>/dev/null)\n"
😊😊😊😊
ip-10-10-0-13# printf "$(dd if=/dev/xbd1 bs=1 count=14 skip=8 2>/dev/null)\n"
😊😊😊😊

ip-10-10-0-13#
14 bytes transferred in 0.001 secs (14000 bytes/sec)

```
ip-10-10-0-13# printf "$(dd if=/dev/xbd1 count=1 2>/dev/null)\n"
```

```
 😐
```

```
ip-10-10-0-13# printf "$(dd if=/dev/xbd1 count=2 2>/dev/null)\n"
```

```
 😤
```

```
ip-10-10-0-13# printf "$(dd if=/dev/xbd1 count=1 skip=1 2>/dev/null)\n"
```

```
 😝
```

```
ip-10-10-0-13# printf "$(dd if=/dev/xbd1 count=1 skip=2 2>/dev/null)\n"
```

```
 🐶❄️
```

```
ip-10-10-0-13# dd if=/dev/xbd1 count=3 | hexdump -C
3+0 records in
3+0 records out
00000000 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ...
00000010 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ...

* 1536 bytes transferred in 0.001 secs (1536000 bytes/sec)
00000200 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ...
00000210 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ...

* 00000400 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ...
00000410 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ...

* 00000600
```

```
Let’s pretend we’re a filesystem…
Let’s pretend we’re a filesystem…

Meta-data (16 bytes total):
• identifier (2 bytes)
• permission (4 bytes)
• owner (1 byte)
• group (1 byte)
• size (4 bytes)
• offset (4 bytes)

data stored in separate area
ip-10-10-0-13#
ip-10-10-0-13#
We don’t have to pretend to be a filesystem…

• The filesystem is responsible for storing the data on the disk.
• To read/write data, it needs to know in which physical blocks the actual data is located.
• Meta data may be separated from file data.
• On a high level, a filesystem really just describes a data storage format.

Next time: the UNIX Filesystem
Links

File Systems and Storage Models:

Understanding Unix Filesystems:
https://is.gd/wGgJOe