

1 Dateisystem-Implementierung (1.440 Punkte)

In dieser Übung sollen Sie das auf dem vorherigen Übungsblatt ausgearbeitete (und gegebenenfalls angepasste) Dateisystem-Design implementieren.

Bei dem darunter liegenden Speichergerät soll es sich sowohl um ein reales Blockgerät als auch beispielsweise um eine Datei in einem anderen Dateisystem handeln können; Sie können eine solche Datei beispielsweise mit `fallocate` oder `truncate` erstellen. Das Speichergerät soll dabei als Argument an Ihr FUSE-Dateisystem übergeben werden können. Der Aufruf könnte dabei wie folgt aussehen:

```
$ ./heafs device mountpoint
```

Stellen Sie sicher, dass Änderungen wirklich auf das Speichergerät geschrieben werden; dies soll spätestens bei einem Aufruf von `fsync` durch die Anwendung geschehen. Das Dateisystem soll außerdem möglichst robust gegen mögliche Abstürze sein.

Hinweis: Um die Implementierung möglichst einfach zu halten, bietet es sich an `mmap` zu nutzen. Dabei kann das gesamte Blockgerät bzw. die gesamte Datei als zusammenhängender Speicherbereich in den Adressraum Ihres Dateisystems eingeblendet werden, selbst wenn das Blockgerät bzw. die Datei größer ist als der zur Verfügung stehende Arbeitsspeicher.

2 Leistungsmessung (90 Punkte)

Analysieren Sie ihr Dateisystem mittels des in den Materialien befindlichen Benchmarks. Erstellen Sie geeignete Diagramme und notieren Sie Ihre Schlussfolgerungen.

Der Benchmark lässt sich wie folgt aufrufen, wobei `$mnt` für den Einhängpunkt Ihres Dateisystems steht:

```
$ ./data --path=$mnt --sync-on-close --iterations=3
```

Weitere Optionen können Sie sich mit `--help` anzeigen lassen. Messen Sie die Leistung Ihrer Implementierung mit der Option `--sync-on-close`, 1-12 Threads (`--threads`) und geeigneten Blockgrößen (`--block-size`) von 4 KiB und 4 MiB. Variieren Sie dabei die Blockanzahl (`--block-count`) so, dass der Benchmark einige Minuten läuft und die maximale Dateigröße von 256 MiB nicht überschritten wird. Führen Sie die Messungen sowohl für thread-lokale Dateien als auch für eine geteilte Datei (`--shared`) durch.

Abgabe

Erstellen Sie ein Verzeichnis mit ihrem C-Programm `heafs.c`, dem dazugehörigen Makefile und der Datei `auswertung.pdf`. Packen Sie ein komprimiertes Archiv aus dem sauberen Verzeichnis (ohne Binärdateien).

Senden Sie das Archiv an `hea-abgabe@wr.informatik.uni-hamburg.de`.