

Opgave 4 OSM

Davy Eskildsen og Mads Lund

March 12, 2012

1 FlatFS

I denne opgave har vi implementeret filsystemet FlatFS som kan ses som en udvidelse af det allerede implementeret TFS. Hvor TFS baserer sig på en inode med pointere til filens datablokke, giver FlatFS mulighed for enkelt og dobbelt indirekte pointere som pejer på en blok indeholdende flere pointere.

Dette medfører forøgelse af den maksimale filstørrelse. FlatFS indeholder 7 direkte pointere til datablokke

$$n_{direcPointers} = 7.$$

Hver indirekte pointer pejer på en blok indeholdende pointere. Antallet er

$$n_{pointers} = \frac{size_{block}}{size_{ref}}.$$

hvor $size_{block}$ er blokstørrelsen og $size_{ref}$ er størrelsen på en reference. Vi kan nu beregne filstørrelsen

$$size_{file} = size_{block} (n_{direcPointers} + n_{pointers} + n_{pointers}^2).$$

Hvis vi antager at blokstørrelsen er 512 byte og referencerne er 4 byte så er

$$n_{pointers} = \frac{512}{4} = 128$$

$$\begin{aligned} size_{file} &= 512 \text{ byte } (7 + 128 + 128^2) \\ &= 5457728 \text{ byte} \\ &\approx 8 \text{ MB} \end{aligned}$$

2 Implementation af flatfs

2.1 Allokering af blokke

Når da FlatFS understøtter dynamisk blokallokering lader vi `write` bruge allokeringsblokken til at allokere blokke mens der skrives. For at undgå at skulle hente allokeringsblokken flere gange gemmer vi den i `flatfs_bat` og bruger i stedet `flatfs_md` som temporær buffer til læsninger fra disken.

3 Status på implementationen

Vi nåede desværre ikke at få fil-systemet op og køre endnu. Vi har skrevet nogle kommentarer i blandt andet flatf.c.