

# Leitura assíncrona de blocos de dados no Nanvix

Gustavo F. Guimarães, Gustavo Zambonin, Marcello Klingelfus

Sistemas Operacionais I (UFSC-INE5412)

## 1 Descrição da técnica

Em um computador, a hierarquia de memória é geralmente organizada considerando a velocidade de acesso de dados dentro do hardware. O armazenamento secundário, em sua maioria discos magnéticos, está no nível mais baixo — longe do processador e, por conta de seu funcionamento essencialmente mecânico, o mais lento dos dispositivos de armazenamento de dados. Por conta dessa característica, é necessário que acessos ao disco sejam reduzidos ou aproveitados ao máximo, a fim de manter a responsividade do sistema.

Para tal, a técnica de *prefetching* pode ser utilizada: consiste da leitura antecipada de dados do disco, de modo assíncrono, para uma estrutura localizada junto ao sistema operacional, chamada de *buffer cache*. É possível observar que, em casos de acesso sequencial a um arquivo, blocos adjacentes ao desejado serão lidos e assim menos faltas serão geradas na cache supracitada, por exemplo.

## 2 Características da implementação

De acordo com a descrição da técnica, é necessário que blocos adjacentes válidos sejam lidos. Assim, a função `src/kernel/fs/file.c:file_read` foi modificada de modo a iterar sobre novos blocos (cuja quantidade é configurada a partir de `include/fs/minix.h:BLOCK_AHEAD`) e, caso estes contêm dados, sejam lidos de modo assíncrono por `src/kernel/fs/buffer.c:bread`. Para que não fosse necessário modificar todas as chamadas deste, uma nova função `abread` foi criada com a lógica considerando um novo argumento, decidindo se a leitura será assíncrona, e a função antiga funciona como um *wrapper*, preservando assim a interface.

Na nova função, é necessário considerar a atualização de um atributo do buffer, mostrando se a leitura será síncrona (`src/kernel/fs/fs.h:BUFFER_SYNC`). Se este for o caso, o fluxo de código continua o mesmo; de modo contrário, o comportamento do escalonador de disco será diferenciado por conta da mudança em `src/kernel/dev/ata/ata.c:ata_readblk`, que leva em conta a *flag* descrita acima para modificar o tipo de requisição (`src/kernel/dev/ata/ata.c:REQ_SYNC`).