NAME

FileCache - keep more files open than the system permits

SYNOPSIS

```
use FileCache;
# or
use FileCache maxopen => 16;


cacheout $mode, $path;
# or
cacheout $path;
print $path @data;


$fh = cacheout $mode, $path;
# or
$fh = cacheout $path;
print $fh @data;
```

DESCRIPTION

The `cacheout` function will make sure that there's a filehandle open for reading or writing available as the pathname you give it. It automatically closes and re-opens files if you exceed your system's maximum number of file descriptors, or the suggested maximum *maxopen*.

cacheout EXPR

> The 1-argument form of cacheout will open a file for writing (`'>'`) on it's first use, and appending (`'>>'`) thereafter.

> Returns EXPR on success for convenience. You may neglect the return value and manipulate EXPR as the filehandle directly if you prefer.

cacheout MODE, EXPR

> The 2-argument form of cacheout will use the supplied mode for the initial and subsequent openings. Most valid modes for 3-argument `open` are supported namely; `'>'`, `'+>'`, `'<'`, `'<+'`, `'>>'`, `'|-'` and `'-|'`

> To pass supplemental arguments to a program opened with `'|-'` or `'-|'` append them to the command string as you would system EXPR.

> Returns EXPR on success for convenience. You may neglect the return value and manipulate EXPR as the filehandle directly if you prefer.

CAVEATS

While it is permissible to `close` a FileCache managed file, do not do so if you are calling `FileCache::cacheout` from a package other than which it was imported, or with another module which overrides `close`. If you must, use `FileCache::cacheout_close`.

Although FileCache can be used with piped opens ('-|' or '|-') doing so is strongly discouraged. If FileCache finds it necessary to close and then reopen a pipe, the command at the far end of the pipe will be reexecuted - the results of performing IO on FileCache'd pipes is unlikely to be what you expect. The ability to use FileCache on pipes may be removed in a future release.

FileCache does not store the current file offset if it finds it necessary to close a file. When the file is reopened, the offset will be as specified by the original `open` file mode. This could be construed to be a bug.

BUGS

sys/param.h lies with its `NOFILE` define on some systems, so you may have to set *maxopen* yourself.