

**NAME**

English - use nice English (or awk) names for ugly punctuation variables

**SYNOPSIS**

```
use English;
use English qw( -no_match_vars ) ; # Avoids regex performance
# penalty in perl 5.16 and
# earlier
...
if ($ERRNO = /denied/) { ... }
```

**DESCRIPTION**

This module provides aliases for the built-in variables whose names no one seems to like to read. Variables with side-effects which get triggered just by accessing them (like \$0) will still be affected.

For those variables that have an **awk** version, both long and short English alternatives are provided. For example, the `$/` variable can be referred to either `$RS` or `$INPUT_RECORD_SEPARATOR` if you are using the English module.

See [perlvar\(1\)](#) for a complete list of these.

**PERFORMANCE**

NOTE: This was fixed in perl 5.20. Mentioning these three variables no longer makes a speed difference. This section still applies if your code is to run on perl 5.18 or earlier.

This module can provoke sizeable inefficiencies for regular expressions, due to unfortunate implementation details. If performance matters in your application and you don't need `$PREMATCH`, `$MATCH`, or `$POSTMATCH`, try doing

```
use English qw( -no_match_vars ) ;
```

. It is especially important to do this in modules to avoid penalizing all applications which use them.