NAME

Pod::PlainText - Convert POD data to formatted ASCII text

SYNOPSIS

```
use Pod::PlainText;
my $parser = Pod::PlainText->new (sentence => 0, width => 78);

# Read POD from STDIN and write to STDOUT.
$parser->parse_from_filehandle;

# Read POD from file.pod and write to file.txt.
$parser->parse_from_file ('file.pod', 'file.txt');
```

DESCRIPTION

NOTE: This module is considered legacy; modern Perl releases (5.18 and higher) are going to remove Pod-Parser from core and use Pod-Simple for all things POD.

Pod::PlainText is a module that can convert documentation in the POD format (the preferred language for documenting Perl) into formatted ASCII. It uses no special formatting controls or codes whatsoever, and its output is therefore suitable for nearly any device.

As a derived class from Pod::Parser, Pod::PlainText supports the same methods and interfaces. See Pod::Parser for all the details; briefly, one creates a new parser with `Pod::PlainText->new()` and then calls either *parse_from_filehandle()* or *parse_from_file()*.

new() can take options, in the form of key/value pairs, that control the behavior of the parser. The currently recognized options are:

alt   If set to a true value, selects an alternate output format that, among other things, uses a different heading style and marks `=item` entries with a colon in the left margin. Defaults to false.

indent
The number of spaces to indent regular text, and the default indentation for `=over` blocks. Defaults to 4.

loose
If set to a true value, a blank line is printed after a `=headN` headings. If set to false (the default), no blank line is printed after `=headN`. This is the default because it's the expected formatting for manual pages; if you're formatting arbitrary text documents, setting this to true may result in more pleasing output.

sentence
If set to a true value, Pod::PlainText will assume that each sentence ends in two spaces, and will try to preserve that spacing. If set to false, all consecutive whitespace in non-verbatim paragraphs is compressed into a single space. Defaults to true.

width
The column at which to wrap text on the right-hand side. Defaults to 76.

The standard Pod::Parser method *parse_from_filehandle()* takes up to two arguments, the first being the file handle to read POD from and the second being the file handle to write the formatted output to. The first defaults to STDIN if not given, and the second defaults to STDOUT. The method *parse_from_file()* is almost identical, except that its two arguments are the input and output disk files instead. See Pod::Parser for the specific details.

DIAGNOSTICS

Bizarre space in item
(W) Something has gone wrong in internal `=item` processing. This message indicates a bug in Pod::PlainText; you should never see it.

Can't open %s for reading: %s
> (F) Pod::PlainText was invoked via the compatibility mode *pod2text()* interface and the input file it was given could not be opened.

Unknown escape: %s
> (W) The POD source contained an `E<>` escape that Pod::PlainText didn't know about.

Unknown sequence: %s
> (W) The POD source contained a non-standard internal sequence (something of the form `X<>`) that Pod::PlainText didn't know about.

Unmatched =back
> (W) Pod::PlainText encountered a `=back` command that didn't correspond to an `=over` command.

RESTRICTIONS

Embedded Ctrl-As (octal 001) in the input will be mapped to spaces on output, due to an internal implementation detail.

NOTES

This is a replacement for an earlier Pod::Text module written by Tom Christiansen. It has a revamped interface, since it now uses Pod::Parser, but an interface roughly compatible with the old *Pod::Text::pod2text()* function is still available. Please change to the new calling convention, though.

The original Pod::Text contained code to do formatting via termcap sequences, although it wasn't turned on by default and it was problematic to get it to work at all. This rewrite doesn't even try to do that, but a subclass of it does. Look for Pod::Text::Termcap.

SEE ALSO

Pod::PlainText is part of the Pod::Parser distribution.

Pod::Parser, Pod::Text::Termcap, *pod2text(1)*

AUTHOR

Please report bugs using <http://rt.cpan.org>.

Russ Allbery <rra@stanford.edu>, based *very* heavily on the original Pod::Text by Tom Christiansen <tchrist@mox.perl.com> and its conversion to Pod::Parser by Brad Appleton <bradapp@enteract.com>.