NAME

Pod::Simple::SimpleTree -- parse Pod into a simple parse tree

SYNOPSIS

```
% cat ptest.pod

=head1 PIE

I like B<pie>!

% perl -MPod::Simple::SimpleTree -MData::Dumper -e \
   "print Dumper(Pod::Simple::SimpleTree->new->parse_file(shift)->root)" \
   ptest.pod

$VAR1 = [
          'Document',
          { 'start_line' => 1 },
          [
            'head1',
            { 'start_line' => 1 },
            'PIE'
          ],
          [
            'Para',
            { 'start_line' => 3 },
            'I like ',
            [
              'B',
              {},
              'pie'
            ],
            '!'
          ]
        ];
```

DESCRIPTION

This class is of interest to people writing a Pod processor/formatter.

This class takes Pod and parses it, returning a parse tree made just of arrayrefs, and hashrefs, and strings.

This is a subclass of *Pod::Simple* and inherits all its methods.

This class is inspired by XML::Parser's "Tree" parsing-style, although it doesn't use exactly the same LoL format.

METHODS

At the end of the parse, call `$parser->root` to get the tree's top node.

Tree Contents

Every element node in the parse tree is represented by an arrayref of the form: [*elementname*, \%attributes, *...subnodes...*]. See the example tree dump in the Synopsis, above.

Every text node in the tree is represented by a simple (non-ref) string scalar. So you can test

`ref($node)` to see whather you have an element node or just a text node.

The top node in the tree is `[ 'Document', \%attributes, ...subnodes... ]`

SEE ALSO

Pod::Simple

perllol

The "Tree" subsubsection in XML::Parser

COPYRIGHT AND DISCLAIMERS

Copyright (c) 2002 Sean M. Burke. All rights reserved.

This library is free software; you can redistribute it and/or modify it under the same terms as Perl itself.

This program is distributed in the hope that it will be useful, but without any warranty; without even the implied warranty of merchantability or fitness for a particular purpose.

AUTHOR

Sean M. Burke `sburke@cpan.org`