NAME

XML::LibXML::PI − XML::LibXML Processing Instructions

SYNOPSIS

```
  use XML::LibXML;
  # Only methods specific to Processing Instruction nodes are listed here,
  # see the XML::LibXML::Node manpage for other methods

  $pinode->setData( $data_string );
  $pinode->setData( name=>string_value [...] );
```

DESCRIPTION

Processing instructions are implemented with XML::LibXML with read and write access. The PI data is the PI without the PI target (as specified in XML 1.0 [17]) as a string. This string can be accessed with getData as implemented in XML::LibXML::Node.

The write access is aware about the fact, that many processing instructions have attribute like data. Therefore **setData()** provides besides the DOM spec conform Interface to pass a set of named parameter. So the code segment

```
  my $pi = $dom->createProcessingInstruction("abc");
  $pi->setData(foo=>'bar', foobar=>'foobar');
  $dom->appendChild( $pi );
```

will result the following PI in the DOM:

```
  <?abc foo="bar" foobar="foobar"?>
```

Which is how it is specified in the DOM specification. This three step interface creates temporary a node in perl space. This can be avoided while using the **insertProcessingInstruction()** method. Instead of the three calls described above, the call

```
  $dom->insertProcessingInstruction("abc",'foo="bar" foobar="foobar"');
```

will have the same result as above.

XML::LibXML::PI's implementation of **setData()** documented below differs a bit from the standard version as available in XML::LibXML::Node:

setData

```
    $pinode->setData( $data_string );
    $pinode->setData( name=>string_value [...] );
```

This method allows one to change the content data of a PI. Additionally to the interface specified for DOM Level2, the method provides a named parameter interface to set the data. This parameter list is converted into a string before it is appended to the PI.

AUTHORS

Matt Sergeant, Christian Glahn, Petr Pajas

VERSION

2.0134

COPYRIGHT

2001−2007, AxKit.com Ltd.

2002−2006, Christian Glahn.

2006−2009, Petr Pajas.

LICENSE

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