PPI

Perl parsen...

Was fällt uns dazu ein?

- Only perl can parse Perl
- Only perl can parse Perl
- Only perl can parse Perl
  - Only perl can parse Perl
    - Only perl can parse Perl
      - Only perl can parse Perl
        - » Only perl can parse Perl

perl can only parse Perl

- Stimmt das?

- Was ist Perl?

Das ist Perl (für perl)

```
use Acme::Buffy;
BUffY bUFFY BUffY bUFFY bUfFy buFfY
bUFfY bUFFY buffy   buffY bUffy buFFY buffY
BuFfybUFfy bufFy BuFFY   buffy bUffy buFfY
buffY BuFfy    buFfybUFFy bUfFY    buffY buFfY
BUfFY BUffy    BuFfY BUFfY BUfFY BuFfy
BuffYbUFfy BuFFY BuFFy   bufFYBUffy
BUfFY BuFfy    BuFFY BUFfy bUfFy buFfy
buffy BUffY bUfFy BUfFY  bu
```

Das ist Perl (für perl)

```
use Acme::DoubleHelix;

 CG
T--A
A---T
A----T
 C----G
  T----A
   A---T
    G--C
     AT
     CG
    C--G
   G---C
```

Das ist Perl (für alle)

```
#!/usr/bin/perl

use strict;
use warnings;
use PPI;

my $ppi = PPI::Document->new($0);
my $dumper = PPI::Dumper->new($ppi);
$dumper->print();
```

Was ist parsen

- Ist für jeden unterschiedlich
  - Zwei Menschen – zwei Meinungen
  - Maschine != Mensch

Was ist parsen

Ein **Parser** (engl. *to parse* „analysieren“ bzw. von lateinisch *pars* „Teil“; weshalb Parser im Deutschen gelegentlich auch als *Zerteiler* bezeichnet werden) ist ein Computerprogramm, das entscheidet, ob ein Eingabetext zur formalen Sprache einer bestimmten Grammatik gehört.

So parst perl

```
#!/usr/bin/perl

use strict;
use warnings;
use PPI;

my $ppi = PPI::Document->new($0);
my $dumper = PPI::Dumper->new($ppi);
$dumper->print();
```

Pfad des Interpreters

So parst perl

```
#!/usr/bin/perl

use strict;
use warnings;
use PPI;

my $ppi = PPI::Document->new($0);
my $dumper = PPI::Dumper->new($ppi);
$dumper->print();
```

So parst perl

```
#!/usr/bin/perl

use strict;
use warnings;
use PPI;

my $ppi = PPI::Document->new($0);
my $dumper = PPI::Dumper->new($ppi);
$dumper->print();
```

Suche strict.pm und führe strict.pm aus – springt aus Programm raus

So parst perl

```
#!/usr/bin/perl

use strict;
use warnings;
use PPI;

my $ppi = PPI::Document->new($0);
my $dumper = PPI::Dumper->new($ppi);
$dumper->print();
```

Suche warnings.pm und führe warnings.pm aus – springt aus Programm raus

So parst perl

```
#!/usr/bin/perl

use strict;
use warnings;
use PPI;

my $ppi = PPI::Document->new($0);
my $dumper = PPI::Dumper->new($ppi);
$dumper->print();
```

Suche PPI.pm und führe PPI.pm aus – springt aus Programm raus

So parst perl

```
#!/usr/bin/perl

use strict;
use warnings;
use PPI;

my $ppi = PPI::Document->new($0);
my $dumper = PPI::Dumper->new($ppi);
$dumper->print();
```

Erzeuge neues PPI::Document-Objekt

Menschliches parsen

- Nur auf ein Dokument bezogen
- Inhalt des Dokuments analysieren

Was ist PPI

- PPI = Perl::Parse::Isolated
- parst Perl-Dokumente nach menschlichem Verständnis
- baut einen Baum auf
  - Ermöglich das Durchwandern für die Suche nach Knoten (Token)
  - Ausgabe über PPI::Dumper
- nützlich zur Quellcode-Analyse

So parst PPI

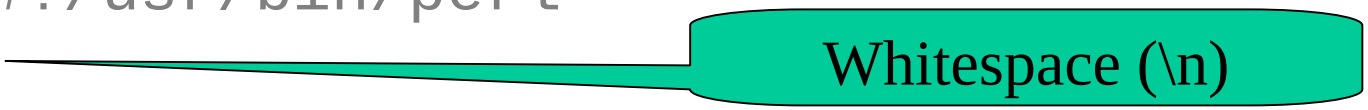

```
#!/usr/bin/perl

use strict;
use warnings;
use PPI;

my $ppi = PPI::Document->new($0);
my $dumper = PPI::Dumper->new($ppi);
$dumper->print();
```

So parst PPI

```
#!/usr/bin/perl

use strict;
use warnings;
use PPI;

my $ppi = PPI::Document->new($0);
my $dumper = PPI::Dumper->new($ppi);
$dumper->print();
```

So parst PPI

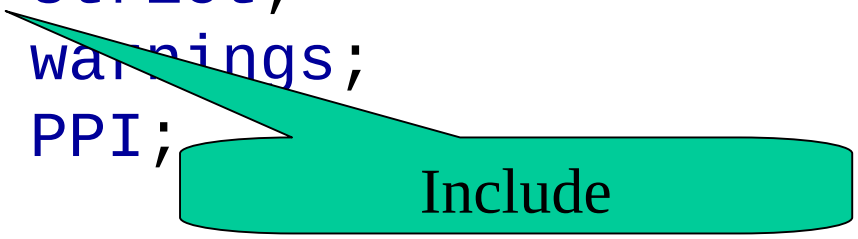
```
#!/usr/bin/perl
```




```
use strict;
use warnings;
use PPI;

my $ppi = PPI::Document->new($0);
my $dumper = PPI::Dumper->new($ppi);
$dumper->print();
```

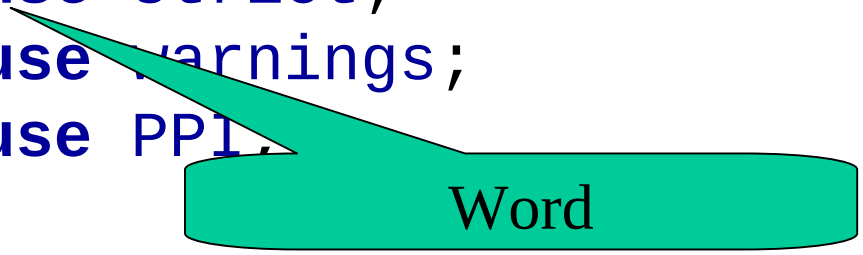
So parst PPI

```
#!/usr/bin/perl

use strict;
use warnings;
use PPI;

my $ppi = PPI::Document->new($0);
my $dumper = PPI::Dumper->new($ppi);
$dumper->print();
```



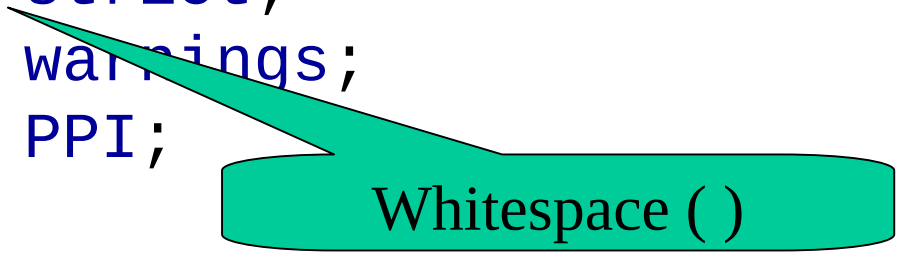
So parst PPI

```
#!/usr/bin/perl

use strict;
use warnings;
use PPI;

my $ppi = PPI::Document->new($0);
my $dumper = PPI::Dumper->new($ppi);
$dumper->print();
```



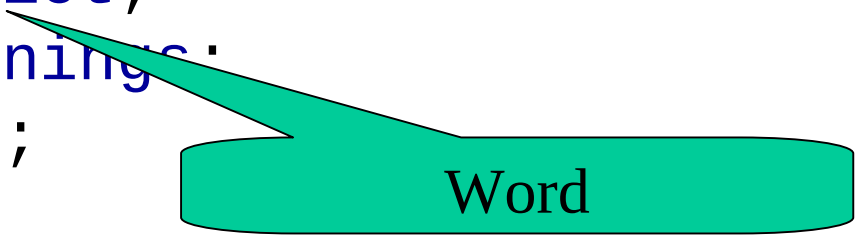
So parst PPI

```
#!/usr/bin/perl

use strict;
use warnings;
use PPI;

my $ppi = PPI::Document->new($0);
my $dumper = PPI::Dumper->new($ppi);
$dumper->print();
```



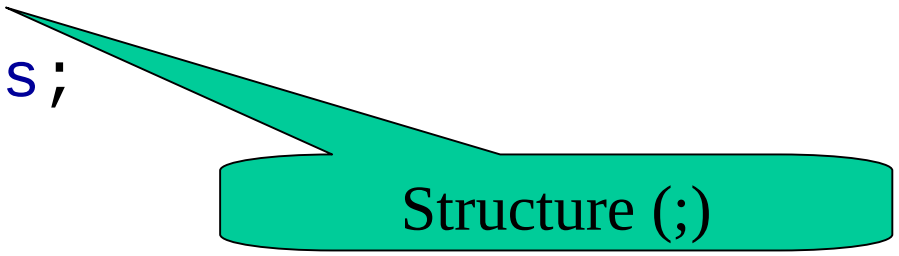
So parst PPI

```
#!/usr/bin/perl

use strict;
use warnings;
use PPI;

my $ppi = PPI::Document->new($0);
my $dumper = PPI::Dumper->new($ppi);
$dumper->print();
```



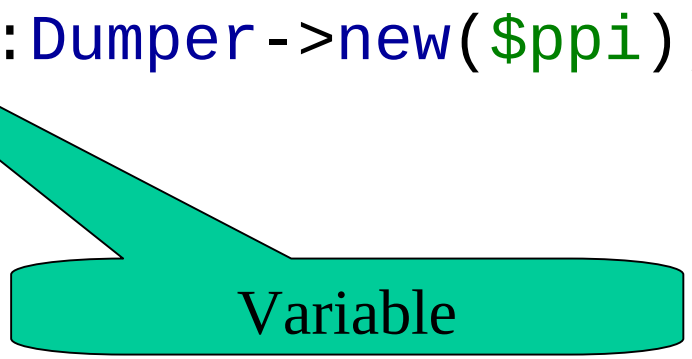
So parst PPI

```
#!/usr/bin/perl

use strict;
use warnings;
use PPI;

my $ppi = PPI::Document->new($0);
my $dumper = PPI::Dumper->new($ppi);
$dumper->print();
```



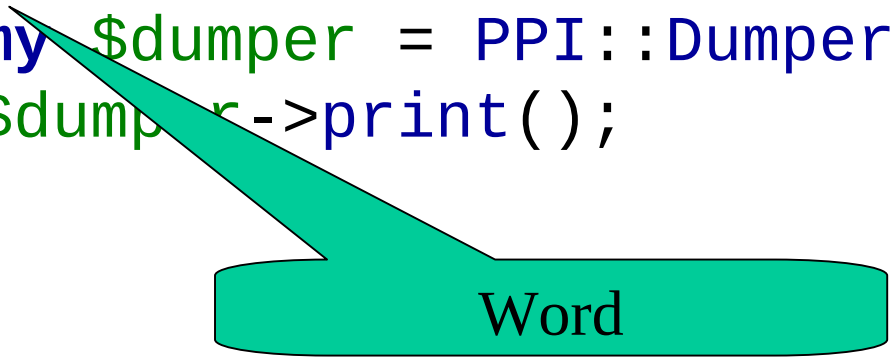
So parst PPI

```
#!/usr/bin/perl

use strict;
use warnings;
use PPI;

my $ppi = PPI::Document->new($0);
my $dumper = PPI::Dumper->new($ppi);
$dumper->print();
```



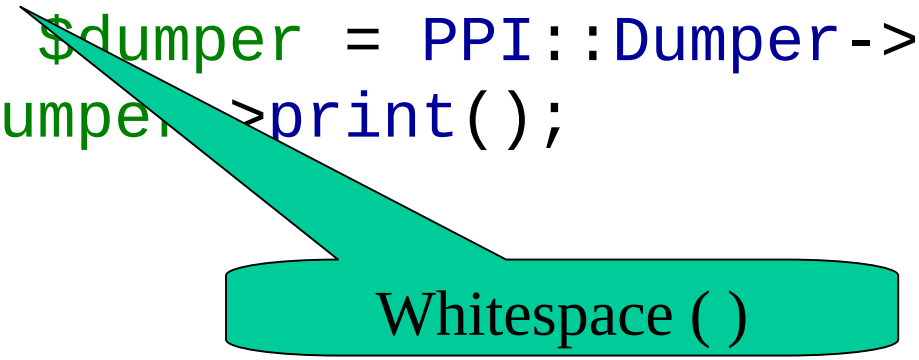
So parst PPI

```
#!/usr/bin/perl

use strict;
use warnings;
use PPI;

my $ppi = PPI::Document->new($0);
my $dumper = PPI::Dumper->new($ppi);
$dumper->print();
```



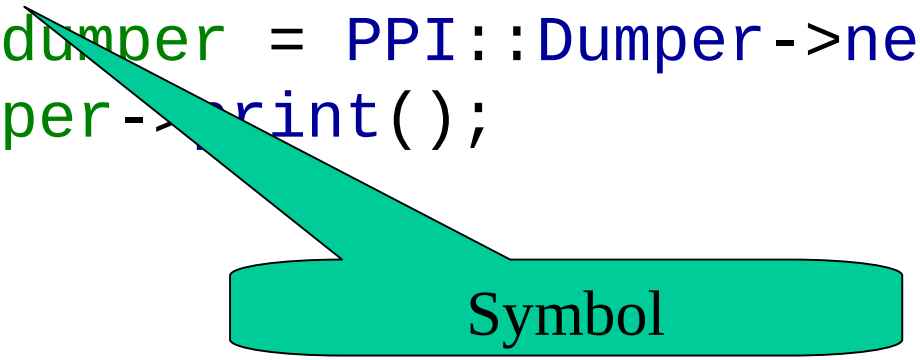
So parst PPI

```
#!/usr/bin/perl

use strict;
use warnings;
use PPI;

my $ppi = PPI::Document->new($0);
my $dumper = PPI::Dumper->new($ppi);
$dumper->print();
```



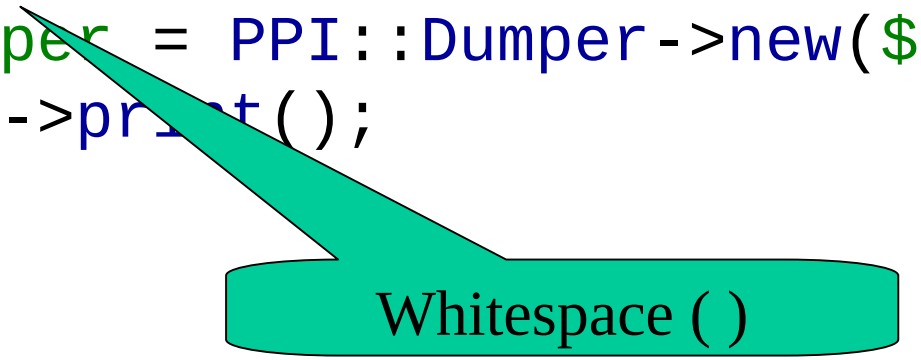
So parst PPI

```
#!/usr/bin/perl

use strict;
use warnings;
use PPI;

my $ppi = PPI::Document->new($0);
my $dumper = PPI::Dumper->new($ppi);
$dumper->print();
```



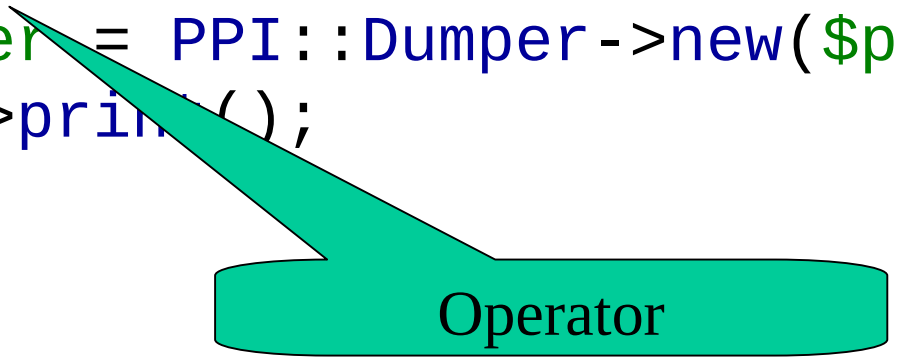
So parst PPI

```
#!/usr/bin/perl

use strict;
use warnings;
use PPI;

my $ppi = PPI::Document->new($0);
my $dumper = PPI::Dumper->new($ppi);
$dumper->print();
```



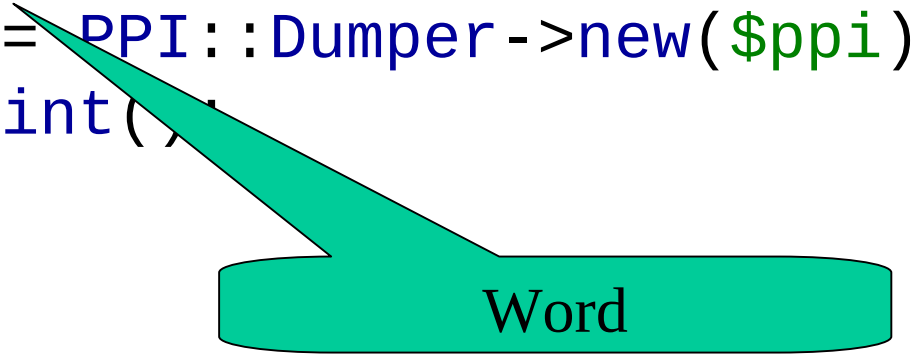
So parst PPI

```
#!/usr/bin/perl

use strict;
use warnings;
use PPI;

my $ppi = PPI::Document->new($0);
my $dumper = PPI::Dumper->new($ppi);
$dumper->print();
```



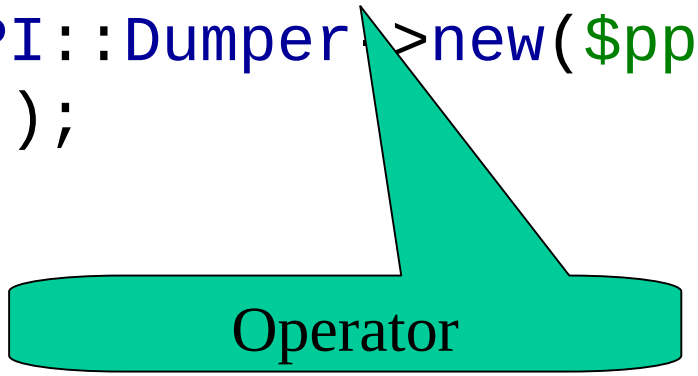
So parst PPI

```
#!/usr/bin/perl

use strict;
use warnings;
use PPI;

my $ppi = PPI::Document->new($0);
my $dumper = PPI::Dumper->new($ppi);
$dumper->print();
```



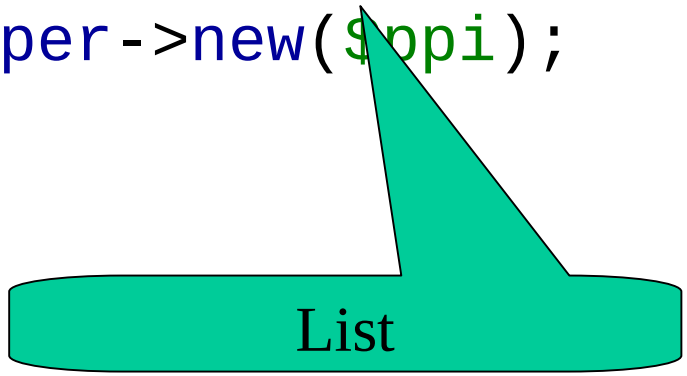
So parst PPI

```
#!/usr/bin/perl

use strict;
use warnings;
use PPI;

my $ppi = PPI::Document->new($0);
my $dumper = PPI::Dumper->new($ppi);
$dumper->print();
```



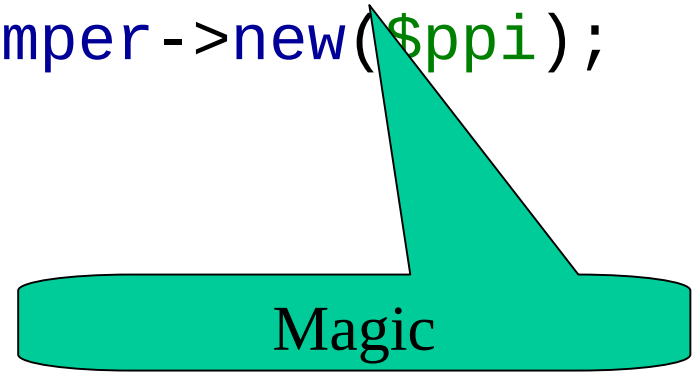
So parst PPI

```
#!/usr/bin/perl

use strict;
use warnings;
use PPI;

my $ppi = PPI::Document->new($0);
my $dumper = PPI::Dumper->new($ppi);
$dumper->print();
```



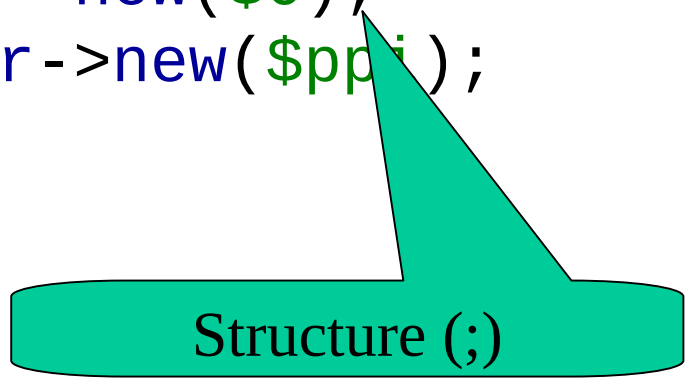
So parst PPI

```
#!/usr/bin/perl

use strict;
use warnings;
use PPI;

my $ppi = PPI::Document->new($0);
my $dumper = PPI::Dumper->new($ppi);
$dumper->print();
```



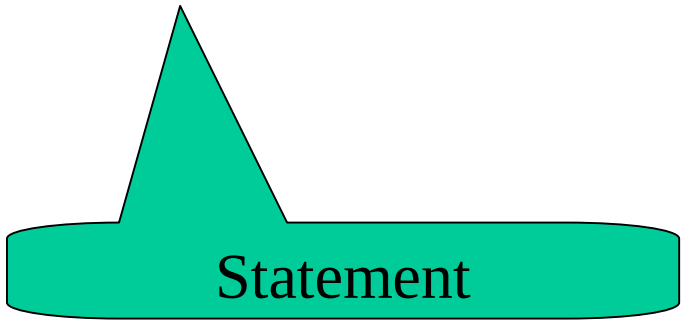
So parst PPI

```
#!/usr/bin/perl

use strict;
use warnings;
use PPI;

my $ppi = PPI::Document->new($0);
my $dumper = PPI::Dumper->new($ppi);
$dumper->print();
```



So parst PPI

```
#!/usr/bin/perl

use strict;
use warnings;
use PPI;

my $ppi = PPI::Document->new($0);
my $dumper = PPI::Dumper->new($ppi);
$dumper->print();
```



So parst PPI das Programm

```
PPI::Document
  PPI::Token::Comment   '#!/usr/bin/perl\n'
  PPI::Token::Whitespace        '\n'
  PPI::Statement::Include
    PPI::Token::Word    'use'
    PPI::Token::Whitespace      ' '
    PPI::Token::Word    'strict'
    PPI::Token::Structure       ';'
  PPI::Token::Whitespace        '\n'
[...]
  PPI::Statement::Variable
    PPI::Token::Word    'my'
    PPI::Token::Whitespace      ' '
    PPI::Token::Symbol          '$ppi'
    PPI::Token::Whitespace      ' '
    PPI::Token::Operator        '='
    PPI::Token::Whitespace      ' '
    PPI::Token::Word    'PPI::Document'
    PPI::Token::Operator        '->'
    PPI::Token::Word    'new'
    PPI::Structure::List        ( ... )
      PPI::Statement::Expression
        PPI::Token::Magic       '$0, [...]
```

Wo findet es Verwendung

- Kephra
- UPC
- Perl::Critic
- PPI::HTML
- Class::Superclasses
- Überall wo die Ausführung nicht erwünscht ist

Weiterführende Links

- http://www.perl.com/pub/a/2005/06/09/ppi.html
- http://search.cpan.org/dist/PPI