

NAME

ExtUtils::ParseXS::Utilities - Subroutines used with ExtUtils::ParseXS

SYNOPSIS

```

use ExtUtils::ParseXS::Utilities qw(
    standard_typemap_locations
    trim_whitespace
    C_string
    valid_proto_string
    process_typemaps
    map_type
    standard_XS_defs
    assign_func_args
    analyze_preprocessor_statements
    set_cond
    Warn
    blurt
    death
    check_conditional_preprocessor_statements
    escape_file_for_line_directive
    report_typemap_failure
);

```

SUBROUTINES

The following functions are not considered to be part of the public interface. They are documented here for the benefit of future maintainers of this module.

`standard_typemap_locations()`

- Purpose

Provide a list of filepaths where *typemap* files may be found. The filepaths — relative paths to files (not just directory paths) — appear in this list in lowest-to-highest priority.

The highest priority is to look in the current directory.

```
'typemap'
```

The second and third highest priorities are to look in the parent of the current directory and a directory called *lib/ExtUtils* underneath the parent directory.

```
'../typemap',
'../lib/ExtUtils/typemap',
```

The fourth through ninth highest priorities are to look in the corresponding grandparent, great-grandparent and great-great-grandparent directories.

```
'../../typemap',
'../../lib/ExtUtils/typemap',
'../../../typemap',
'../../../lib/ExtUtils/typemap',
'../../../../../typemap',
'../../../../../lib/ExtUtils/typemap',
```

The tenth and subsequent priorities are to look in directories named *ExtUtils* which are subdirectories of directories found in `@INC` — *provided* a file named *typemap* actually exists in such a directory. Example:

```
'/usr/local/lib/perl5/5.10.1/ExtUtils/typemap',
```

However, these filepaths appear in the list returned by `standard_typemap_locations()` in reverse order, *i.e.*, lowest-to-highest.

```

'/usr/local/lib/perl5/5.10.1/ExtUtils/typemap',
'../../../../lib/ExtUtils/typemap',
'../../../../typemap',
'../../../../lib/ExtUtils/typemap',
'../../../../typemap',
'../../lib/ExtUtils/typemap',
'../../typemap',
'../lib/ExtUtils/typemap',
'../typemap',
'typemap'

```

- Arguments

```
my @stl = standard_typemap_locations( \@INC );
```

Reference to @INC.

- Return Value

Array holding list of directories to be searched for *typemap* files.

trim_whitespace()

- Purpose

Perform an in-place trimming of leading and trailing whitespace from the first argument provided to the function.

- Argument

```
trim_whitespace($arg);
```

- Return Value

None. Remember: this is an *in-place* modification of the argument.

C_string()

- Purpose

Escape backslashes (\) in prototype strings.

- Arguments

```
$ProtoThisXSUB = C_string($_);
```

String needing escaping.

- Return Value

Properly escaped string.

valid_proto_string()

- Purpose

Validate prototype string.

- Arguments

String needing checking.

- Return Value

Upon success, returns the same string passed as argument.

Upon failure, returns 0.

process_typemaps()

- Purpose

Process all typemap files.

- Arguments


```
my $typemaps_object = process_typemaps( $args{typemap}, $pwd );
```

 List of two elements: `typemap` element from `%args`; current working directory.
- Return Value

Upon success, returns an [ExtUtils::Typemaps](#) object.

map_type()

- Purpose

Performs a mapping at several places inside PARAGRAPH loop.
- Arguments


```
$type = map_type($self, $type, $varname);
```

 List of three arguments.
- Return Value

String holding augmented version of second argument.

standard_XS_defs()

- Purpose

Writes to the `.c` output file certain preprocessor directives and function headers needed in all such files.
- Arguments

None.
- Return Value

Returns true.

assign_func_args()

- Purpose

Perform assignment to the `func_args` attribute.
- Arguments


```
$string = assign_func_args($self, $argsref, $class);
```

 List of three elements. Second is an array reference; third is a string.
- Return Value

String.

analyze_preprocessor_statements()

- Purpose

Within each function inside each Xsub, print to the `.c` output file certain preprocessor statements.
- Arguments


```
( $self, $XSS_work_idx, $BootCode_ref ) =  
analyze_preprocessor_statements(  
$self, $statement, $XSS_work_idx, $BootCode_ref  
);
```

 List of four elements.
- Return Value

Modified values of three of the arguments passed to the function. In particular, the `XSSStack` and `InitFileCode` attributes are modified.

`set_cond()`

- Purpose
- Arguments
- Return Value

`current_line_number()`

- Purpose

Figures out the current line number in the XS file.
- Arguments

`$self`
- Return Value

The current line number.

`Warn()`

- Purpose
- Arguments
- Return Value

`blurt()`

- Purpose
- Arguments
- Return Value

`death()`

- Purpose
- Arguments
- Return Value

`check_conditional_preprocessor_statements()`

- Purpose
- Arguments
- Return Value

`escape_file_for_line_directive()`

- Purpose

Escapes a given code source name (typically a file name but can also be a command that was read from) so that double-quotes and backslashes are escaped.
- Arguments

A string.
- Return Value

A string with escapes for double-quotes and backslashes.

`report_typemap_failure`

- Purpose

Do error reporting for missing typemaps.
- Arguments

The `ExtUtils::ParseXS` object.

An `ExtUtils::Typemaps` object.

The string that represents the C type that was not found in the typemap.

Optionally, the string `death` or `blurt` to choose whether the error is immediately fatal or not.
Default: `blurt`

- Return Value

Returns nothing. Depending on the arguments, this may call `death` or `blurt`, the former of which is fatal.