NAME

Devel::PPPort - Perl/Pollution/Portability

SYNOPSIS

```
Devel::PPPort::WriteFile();     # defaults to ./ppport.h
Devel::PPPort::WriteFile('someheader.h');
```

DESCRIPTION

Perl's API has changed over time, gaining new features, new functions, increasing its flexibility, and reducing the impact on the C namespace environment (reduced pollution). The header file written by this module, typically *ppport.h*, attempts to bring some of the newer Perl API features to older versions of Perl, so that you can worry less about keeping track of old releases, but users can still reap the benefit.

`Devel::PPPort` contains a single function, called `WriteFile`. Its only purpose is to write the *ppport.h* C header file. This file contains a series of macros and, if explicitly requested, functions that allow XS modules to be built using older versions of Perl. Currently, Perl versions from 5.003 to 5.10.0 are supported.

This module is used by `h2xs` to write the file *ppport.h*.

Why use ppport.h?

You should use *ppport.h* in modern code so that your code will work with the widest range of Perl interpreters possible, without significant additional work.

You should attempt older code to fully use *ppport.h*, because the reduced pollution of newer Perl versions is an important thing. It's so important that the old polluting ways of original Perl modules will not be supported very far into the future, and your module will almost certainly break! By adapting to it now, you'll gain compatibility and a sense of having done the electronic ecology some good.

How to use ppport.h

Don't direct the users of your module to download `Devel::PPPort`. They are most probably no XS writers. Also, don't make *ppport.h* optional. Rather, just take the most recent copy of *ppport.h* that you can find (e.g. by generating it with the latest `Devel::PPPort` release from CPAN), copy it into your project, adjust your project to use it, and distribute the header along with your module.

Running ppport.h

But *ppport.h* is more than just a C header. It's also a Perl script that can check your source code. It will suggest hints and portability notes, and can even make suggestions on how to change your code. You can run it like any other Perl program:

```
perl ppport.h [options] [files]
```

It also has embedded documentation, so you can use

```
perldoc ppport.h
```

to find out more about how to use it.

FUNCTIONS

WriteFile

`WriteFile` takes one optional argument. When called with one argument, it expects to be passed a filename. When called with no arguments, it defaults to the filename *ppport.h*.

The function returns a true value if the file was written successfully. Otherwise it returns a false value.

COMPATIBILITY

ppport.h supports Perl versions from 5.003 to 5.10.0 in threaded and non-threaded configurations.

Provided Perl compatibility API

The header file written by this module, typically *ppport.h*, provides access to the following elements of the Perl API that is not available in older Perl releases:

```
_aMY_CXT
_pMY_CXT
aMY_CXT
aMY_CXT_
aTHX
aTHX_
aTHXR
aTHXR_
AvFILLp
boolSV
call_argv
call_method
call_pv
call_sv
ckWARN
CopFILE
CopFILE_set
CopFILEAV
CopFILEGV
CopFILEGV_set
CopFILESV
CopSTASH
CopSTASH_eq
CopSTASH_set
CopSTASHPV
CopSTASHPV_set
CopyD
dAX
dAXMARK
DEFSV
dITEMS
dMY_CXT
dMY_CXT_SV
dNOOP
dTHR
dTHX
dTHXa
dTHXoa
dTHXR
dUNDERBAR
dVAR
dXCPT
dXSTARG
END_EXTERN_C
ERRSV
eval_pv
eval_sv
EXTERN_C
get_av
```

```
get_cv
get_hv
get_sv
grok_bin
grok_hex
grok_number
GROK_NUMERIC_RADIX
grok_numeric_radix
grok_oct
gv_stashpvn
hv_fetchs
hv_stores
IN_LOCALE
IN_LOCALE_COMPILETIME
IN_LOCALE_RUNTIME
IN_PERL_COMPILETIME
INT2PTR
IS_NUMBER_GREATER_THAN_UV_MAX
IS_NUMBER_IN_UV
IS_NUMBER_INFINITY
IS_NUMBER_NAN
IS_NUMBER_NEG
IS_NUMBER_NOT_INT
IVdf
IVSIZE
IVTYPE
load_module
memEQ
memNE
MoveD
mPUSHi
mPUSHn
mPUSHp
mPUSHu
mXPUSHi
mXPUSHn
mXPUSHp
mXPUSHu
MY_CXT
MY_CXT_CLONE
MY_CXT_INIT
my_snprintf
my_strlcat
my_strlcpy
newCONSTSUB
newRV_inc
newRV_noinc
newSVpvn
newSVpvn_share
newSVpvs
newSVuv
Newx
Newxc
Newxz
NOOP
NUM2PTR
```

```
NVef
NVff
NVgf
NVTYPE
packWARN
PERL_ABS
PERL_BCDVERSION
PERL_GCC_BRACE_GROUPS_FORBIDDEN
PERL_HASH
PERL_INT_MAX
PERL_INT_MIN
PERL_LONG_MAX
PERL_LONG_MIN
PERL_MAGIC_arylen
PERL_MAGIC_backref
PERL_MAGIC_bm
PERL_MAGIC_collxfrm
PERL_MAGIC_dbfile
PERL_MAGIC_dbline
PERL_MAGIC_defelem
PERL_MAGIC_env
PERL_MAGIC_envelem
PERL_MAGIC_ext
PERL_MAGIC_fm
PERL_MAGIC_glob
PERL_MAGIC_isa
PERL_MAGIC_isaelem
PERL_MAGIC_mutex
PERL_MAGIC_nkeys
PERL_MAGIC_overload
PERL_MAGIC_overload_elem
PERL_MAGIC_overload_table
PERL_MAGIC_pos
PERL_MAGIC_qr
PERL_MAGIC_regdata
PERL_MAGIC_regdatum
PERL_MAGIC_regex_global
PERL_MAGIC_shared
PERL_MAGIC_shared_scalar
PERL_MAGIC_sig
PERL_MAGIC_sigelem
PERL_MAGIC_substr
PERL_MAGIC_sv
PERL_MAGIC_taint
PERL_MAGIC_tied
PERL_MAGIC_tiedelem
PERL_MAGIC_tiedscalar
PERL_MAGIC_utf8
PERL_MAGIC_uvar
PERL_MAGIC_uvar_elem
PERL_MAGIC_vec
PERL_MAGIC_vstring
PERL_QUAD_MAX
PERL_QUAD_MIN
PERL_REVISION
PERL_SCAN_ALLOW_UNDERSCORES
```

```
PERL_SCAN_DISALLOW_PREFIX
PERL_SCAN_GREATER_THAN_UV_MAX
PERL_SCAN_SILENT_ILLDIGIT
PERL_SHORT_MAX
PERL_SHORT_MIN
PERL_SIGNALS_UNSAFE_FLAG
PERL_SUBVERSION
PERL_UCHAR_MAX
PERL_UCHAR_MIN
PERL_UINT_MAX
PERL_UINT_MIN
PERL_ULONG_MAX
PERL_ULONG_MIN
PERL_UNUSED_ARG
PERL_UNUSED_CONTEXT
PERL_UNUSED_DECL
PERL_UNUSED_VAR
PERL_UQUAD_MAX
PERL_UQUAD_MIN
PERL_USE_GCC_BRACE_GROUPS
PERL_USHORT_MAX
PERL_USHORT_MIN
PERL_VERSION
Perl_warner
Perl_warner_nocontext
PL_compiling
PL_copline
PL_curcop
PL_curstash
PL_DBsignal
PL_DBsingle
PL_DBsub
PL_DBtrace
PL_debstash
PL_defgv
PL_diehook
PL_dirty
PL_dowarn
PL_errgv
PL_expect
PL_hexdigit
PL_hints
PL_laststatval
PL_na
PL_no_modify
PL_perl_destruct_level
PL_perldb
PL_ppaddr
PL_rsfp
PL_rsfp_filters
PL_signals
PL_stack_base
PL_stack_sp
PL_statcache
PL_stdingv
PL_Sv
```

```
PL_sv_arenaroot
PL_sv_no
PL_sv_undef
PL_sv_yes
PL_tainted
PL_tainting
pMY_CXT
pMY_CXT_
Poison
PoisonFree
PoisonNew
PoisonWith
pTHX
pTHX_
PTR2IV
PTR2NV
PTR2ul
PTR2UV
PTRV
PUSHmortal
PUSHu
SAVE_DEFSV
START_EXTERN_C
START_MY_CXT
STMT_END
STMT_START
STR_WITH_LEN
sv_2pv_flags
sv_2pv_nolen
sv_2pvbyte
sv_2pvbyte_nolen
sv_2uv
sv_catpv_mg
sv_catpvf_mg
sv_catpvf_mg_nocontext
sv_catpvn_mg
sv_catpvn_nomg
sv_catpvs
sv_catsv_mg
sv_catsv_nomg
SV_CONST_RETURN
SV_COW_DROP_PV
SV_COW_SHARED_HASH_KEYS
SV_GMAGIC
SV_HAS_TRAILING_NUL
SV_IMMEDIATE_UNREF
sv_magic_portable
SV_MUTABLE_RETURN
SV_NOSTEAL
sv_pvn_force_flags
sv_pvn_nomg
sv_setiv_mg
sv_setnv_mg
sv_setpv_mg
sv_setpvf_mg
sv_setpvf_mg_nocontext
```

```
sv_setpvn_mg
sv_setpvs
sv_setsv_mg
sv_setsv_nomg
sv_setuv
sv_setuv_mg
SV_SMAGIC
sv_usepvn_mg
SV_UTF8_NO_ENCODING
sv_uv
sv_vcatpvf
sv_vcatpvf_mg
sv_vsetpvf
sv_vsetpvf_mg
SVf
SvGETMAGIC
SvIV_nomg
SvMAGIC_set
SvPV_const
SvPV_flags
SvPV_flags_const
SvPV_flags_const_nolen
SvPV_flags_mutable
SvPV_force
SvPV_force_flags
SvPV_force_flags_mutable
SvPV_force_flags_nolen
SvPV_force_mutable
SvPV_force_nolen
SvPV_force_nomg
SvPV_force_nomg_nolen
SvPV_mutable
SvPV_nolen
SvPV_nolen_const
SvPV_nomg
SvPV_nomg_const
SvPV_nomg_const_nolen
SvPVbyte
SvPVX_const
SvPVX_mutable
SvREFCNT_inc
SvREFCNT_inc_NN
SvREFCNT_inc_simple
SvREFCNT_inc_simple_NN
SvREFCNT_inc_simple_void
SvREFCNT_inc_simple_void_NN
SvREFCNT_inc_void
SvREFCNT_inc_void_NN
SvRV_set
SvSHARED_HASH
SvSTASH_set
SvUOK
SvUV
SvUV_nomg
SvUV_set
SvUVX
```

```
SvUVx
SvUVXx
SvVSTRING_mg
UNDERBAR
UTF8_MAXBYTES
UVof
UVSIZE
UVTYPE
UVuf
UVXf
UVxf
vload_module
vnewSVpvf
WARN_ALL
WARN_AMBIGUOUS
WARN_ASSERTIONS
WARN_BAREWORD
WARN_CLOSED
WARN_CLOSURE
WARN_DEBUGGING
WARN_DEPRECATED
WARN_DIGIT
WARN_EXEC
WARN_EXITING
WARN_GLOB
WARN_INPLACE
WARN_INTERNAL
WARN_IO
WARN_LAYER
WARN_MALLOC
WARN_MISC
WARN_NEWLINE
WARN_NUMERIC
WARN_ONCE
WARN_OVERFLOW
WARN_PACK
WARN_PARENTHESIS
WARN_PIPE
WARN_PORTABLE
WARN_PRECEDENCE
WARN_PRINTF
WARN_PROTOTYPE
WARN_QW
WARN_RECURSION
WARN_REDEFINE
WARN_REGEXP
WARN_RESERVED
WARN_SEMICOLON
WARN_SEVERE
WARN_SIGNAL
WARN_SUBSTR
WARN_SYNTAX
WARN_TAINT
WARN_THREADS
WARN_UNINITIALIZED
WARN_UNOPENED
```

```
WARN_UNPACK
WARN_UNTIE
WARN_UTF8
WARN_VOID
warner
XCPT_CATCH
XCPT_RETHROW
XCPT_TRY_END
XCPT_TRY_START
XPUSHmortal
XPUSHu
XSprePUSH
XSRETURN
XSRETURN_UV
XST_mUV
ZeroD
```

Perl API not supported by ppport.h

There is still a big part of the API not supported by *ppport.h*. Either because it doesn't make sense to back-port that part of the API, or simply because it hasn't been implemented yet. Patches welcome!

Here's a list of the currently unsupported API, and also the version of Perl below which it is unsupported:

perl 5.9.5

```
MULTICALL
POP_MULTICALL
PUSH_MULTICALL
Perl_signbit
SvRX
SvRXOK
av_create_and_push
av_create_and_unshift_one
get_cvn_flags
gv_fetchfile_flags
hv_assert
mro_get_linear_isa
mro_get_linear_isa_c3
mro_get_linear_isa_dfs
mro_method_changed_in
my_dirfd
newSV_type
pad_sv
pregcomp
ptr_table_clear
ptr_table_fetch
ptr_table_free
ptr_table_new
ptr_table_split
ptr_table_store
re_compile
re_intuit_start
reg_named_buff_all
reg_named_buff_exists
reg_named_buff_fetch
reg_named_buff_firstkey
```

```
reg_named_buff_nextkey
reg_named_buff_scalar
reg_stringify
regfree_internal
savesharedpvn
scan_vstring
stashpv_hvname_match
upg_version
```

perl 5.9.4

```
PerlIO_context_layers
gv_name_set
my_vsnprintf
newXS_flags
pv_escape
pv_pretty
regclass_swash
sv_does
sv_usepvn_flags
```

perl 5.9.3

```
av_arylen_p
ckwarn
ckwarn_d
csighandler
dMULTICALL
doref
gv_const_sv
gv_stashpvs
hv_eiter_p
hv_eiter_set
hv_name_set
hv_placeholders_get
hv_placeholders_p
hv_placeholders_set
hv_riter_p
hv_riter_set
is_utf8_string_loclen
my_sprintf
newGIVENOP
newSVhek
newSVpvs_share
newWHENOP
newWHILEOP
ref
savepvs
sortsv_flags
vverify
```

perl 5.9.2

```
SvPVbyte_force
find_rundefsvoffset
gv_fetchpvn_flags
gv_fetchsv
```

```
op_refcnt_lock
op_refcnt_unlock
savesvpv
vnormal
```

perl 5.9.1

```
hv_clear_placeholders
hv_scalar
scan_version
sv_2iv_flags
sv_2uv_flags
```

perl 5.9.0

```
new_version
save_set_svflags
vcmp
vnumify
vstringify
```

perl 5.8.3

```
SvIsCOW
SvIsCOW_shared_hash
```

perl 5.8.1

```
SvVOK
doing_taint
find_runcv
is_utf8_string_loc
packlist
save_bool
savestack_grow_cnt
seed
sv_cat_decode
sv_compile_2op
sv_setpviv
sv_setpviv_mg
unpackstring
```

perl 5.8.0

```
hv_iternext_flags
hv_store_flags
is_utf8_idcont
nothreadhook
```

perl 5.7.3

```
PerlIO_clearerr
PerlIO_close
PerlIO_eof
PerlIO_error
PerlIO_fileno
PerlIO_fill
PerlIO_flush
```

```
PerlIO_get_base
PerlIO_get_bufsiz
PerlIO_get_cnt
PerlIO_get_ptr
PerlIO_read
PerlIO_seek
PerlIO_set_cnt
PerlIO_set_ptrcnt
PerlIO_setlinebuf
PerlIO_stderr
PerlIO_stdin
PerlIO_stdout
PerlIO_tell
PerlIO_unread
PerlIO_write
SvLOCK
SvSHARE
SvUNLOCK
atfork_lock
atfork_unlock
custom_op_desc
custom_op_name
deb
debstack
debstackptrs
gv_fetchmeth_autoload
ibcmp_utf8
my_fork
my_socketpair
pack_cat
perl_destruct
pv_uni_display
save_shared_pvref
savesharedpv
sortsv
sv_copypv
sv_magicext
sv_nolocking
sv_nosharing
sv_recode_to_utf8
sv_uni_display
to_uni_fold
to_uni_lower
to_uni_title
to_uni_upper
to_utf8_case
to_utf8_fold
to_utf8_lower
to_utf8_title
to_utf8_upper
unpack_str
uvchr_to_utf8_flags
uvuni_to_utf8_flags
vdeb
```

perl 5.7.2

```
calloc
getcwd_sv
init_tm
malloc
mfree
mini_mktime
my_atof2
my_strftime
op_null
realloc
sv_catpvn_flags
sv_catsv_flags
sv_setsv_flags
sv_utf8_upgrade_flags
swash_fetch
```

perl 5.7.1

```
POPpbytex
bytes_from_utf8
despatch_signals
do_openn
gv_handler
is_lvalue_sub
my_popen_list
save_mortalizesv
save_padsv
scan_num
sv_force_normal_flags
sv_setref_uv
sv_unref_flags
sv_utf8_upgrade
utf8_length
utf8_to_uvchr
utf8_to_uvuni
utf8n_to_uvuni
uvuni_to_utf8
```

perl 5.6.1

```
SvGAMAGIC
apply_attrs_string
bytes_to_utf8
gv_efullname4
gv_fullname4
is_utf8_string
save_generic_pvref
utf16_to_utf8
utf16_to_utf8_reversed
utf8_to_bytes
```

perl 5.6.0

```
SvIOK_UV
SvIOK_notUV
SvIOK_only_UV
SvPOK_only_UTF8
```

```
SvPVbyte_nolen
SvPVbytex
SvPVbytex_force
SvPVutf8
SvPVutf8_force
SvPVutf8_nolen
SvPVutf8x
SvPVutf8x_force
SvUOK
SvUTF8
SvUTF8_off
SvUTF8_on
av_delete
av_exists
call_atexit
cast_i32
cast_iv
cast_ulong
cast_uv
do_gv_dump
do_gvgv_dump
do_hv_dump
do_magic_dump
do_op_dump
do_open9
do_pmop_dump
do_sv_dump
dump_all
dump_eval
dump_form
dump_indent
dump_packsubs
dump_sub
dump_vindent
get_context
get_ppaddr
gv_dump
init_i18nl10n
init_i18nl14n
is_uni_alnum
is_uni_alnum_lc
is_uni_alnumc
is_uni_alnumc_lc
is_uni_alpha
is_uni_alpha_lc
is_uni_ascii
is_uni_ascii_lc
is_uni_cntrl
is_uni_cntrl_lc
is_uni_digit
is_uni_digit_lc
is_uni_graph
is_uni_graph_lc
is_uni_idfirst
is_uni_idfirst_lc
is_uni_lower
```

```
is_uni_lower_lc
is_uni_print
is_uni_print_lc
is_uni_punct
is_uni_punct_lc
is_uni_space
is_uni_space_lc
is_uni_upper
is_uni_upper_lc
is_uni_xdigit
is_uni_xdigit_lc
is_utf8_alnum
is_utf8_alnumc
is_utf8_alpha
is_utf8_ascii
is_utf8_char
is_utf8_cntrl
is_utf8_digit
is_utf8_graph
is_utf8_idfirst
is_utf8_lower
is_utf8_mark
is_utf8_print
is_utf8_punct
is_utf8_space
is_utf8_upper
is_utf8_xdigit
magic_dump
mess
my_atof
my_fflush_all
newANONATTRSUB
newATTRSUB
newXS
newXSproto
new_collate
new_ctype
new_numeric
op_dump
perl_parse
pmop_dump
pv_display
re_intuit_string
reginitcolors
require_pv
safesyscalloc
safesysfree
safesysmalloc
safesysrealloc
save_I8
save_alloc
save_destructor
save_destructor_x
save_re_context
save_vptr
scan_bin
```

```
set_context
set_numeric_local
set_numeric_radix
set_numeric_standard
str_to_version
sv_2pvutf8
sv_2pvutf8_nolen
sv_force_normal
sv_len_utf8
sv_pos_b2u
sv_pos_u2b
sv_pv
sv_pvbyte
sv_pvbyten
sv_pvbyten_force
sv_pvutf8
sv_pvutf8n
sv_pvutf8n_force
sv_rvweaken
sv_utf8_decode
sv_utf8_downgrade
sv_utf8_encode
swash_init
tmps_grow
to_uni_lower_lc
to_uni_title_lc
to_uni_upper_lc
utf8_distance
utf8_hop
vcroak
vform
vmess
vwarn
vwarner
```

perl 5.005_03

```
POPpx
get_vtbl
save_generic_svref
```

perl 5.005

```
PL_modglobal
cx_dump
debop
debprofdump
fbm_compile
fbm_instr
get_op_descs
get_op_names
init_stacks
mg_length
mg_size
newHVhv
new_stackinfo
regdump
```

```
regexec_flags
regnext
runops_debug
runops_standard
save_hints
save_iv
screaminstr
sv_iv
sv_nv
sv_peek
sv_true
```

perl 5.004_05

```
do_binmode
save_aelem
save_helem
```

perl 5.004

```
GIMME_V
G_VOID
HEf_SVKEY
HeHASH
HeKEY
HeKLEN
HePV
HeSVKEY
HeSVKEY_force
HeSVKEY_set
HeVAL
SvSetMagicSV
SvSetMagicSV_nosteal
SvSetSV_nosteal
SvTAINTED
SvTAINTED_off
SvTAINTED_on
block_gimme
call_list
cv_const_sv
delimcpy
do_open
form
gv_autoload4
gv_efullname3
gv_fetchmethod_autoload
gv_fullname3
hv_delayfree_ent
hv_delete_ent
hv_exists_ent
hv_fetch_ent
hv_free_ent
hv_iterkeysv
hv_ksplit
hv_store_ent
ibcmp_locale
my_failure_exit
```

```
my_memcmp
my_pclose
my_popen
newSVpvf
rsignal
rsignal_state
save_I16
save_gp
share_hek
start_subparse
sv_catpvf
sv_catpvf_mg
sv_cmp_locale
sv_derived_from
sv_gets
sv_magic_portable
sv_setpvf
sv_setpvf_mg
sv_taint
sv_tainted
sv_untaint
sv_vcatpvf
sv_vcatpvf_mg
sv_vcatpvfn
sv_vsetpvf
sv_vsetpvf_mg
sv_vsetpvfn
unsharepvn
vnewSVpvf
warner
```

BUGS

If you find any bugs, `Devel::PPPort` doesn't seem to build on your system or any of its tests fail, please use the CPAN Request Tracker at *http://rt.cpan.org/* to create a ticket for the module.

AUTHORS

- Version 1.x of Devel::PPPort was written by Kenneth Albanowski.

- Version 2.x was ported to the Perl core by Paul Marquess.

- Version 3.x was ported back to CPAN by Marcus Holland-Moritz.

COPYRIGHT

Version 3.x, Copyright (C) 2004-2007, Marcus Holland-Moritz.

Version 2.x, Copyright (C) 2001, Paul Marquess.

Version 1.x, Copyright (C) 1999, Kenneth Albanowski.

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

SEE ALSO

See *h2xs*, *ppport.h*.