

1 APR::Const - Perl Interface for APR Constants

1.1 Synopsis

```
# make the constants available but don't import them
use APR::Const -compile => qw(constant names ...);

# w/o the => syntax sugar
use APR::Const ("-compile", qw(constant names ...));

# compile and import the constants
use APR::Const qw(constant names ...);
```

1.2 Description

This package contains constants specific to APR features.

Refer to the `Apache2::Const` description section for more information.

1.3 Constants

1.3.1 *:common*

```
use APR::Const -compile => qw(:common);
```

The `:common` group is for XXX constants.

1.3.1.1 `APR::Const::SUCCESS`

- since: 2.0.00

1.3.2 *:error*

```
use APR::Const -compile => qw(:error);
```

The `:error` group is for XXX constants.

1.3.2.1 `APR::Const::EABOVEROOT`

- since: 2.0.00

1.3.2.2 `APR::Const::EABSOLUTE`

- since: 2.0.00

1.3.2.3 APR::Const::EACCES

- since: 2.0.00

Due to possible variants in conditions matching EACCES, for checking error codes against this you most likely want to use the `APR::Status::is_EACCES` function instead.

1.3.2.4 APR::Const::EAGAIN

- since: 2.0.00

The error *Resource temporarily unavailable*, may be returned by many different system calls, especially IO calls. Most likely you want to use the `APR::Status::is_EAGAIN` function instead.

1.3.2.5 APR::Const::EBADDATE

- since: 2.0.00

1.3.2.6 APR::Const::EBADF

- since: 2.0.00

1.3.2.7 APR::Const::EBADIP

- since: 2.0.00

1.3.2.8 APR::Const::EBADMASK

- since: 2.0.00

1.3.2.9 APR::Const::EBADPATH

- since: 2.0.00

1.3.2.10 APR::Const::EBUSY

- since: 2.0.00

1.3.2.11 APR::Const::ECONNABORTED

- since: 2.0.00

Due to possible variants in conditions matching ECONNABORTED, for checking error codes against this you most likely want to use the `APR::Status::is_ECONNABORTED` function instead.

1.3.2.12 APR::Const::ECONNREFUSED

- since: 2.0.00

1.3.2.13 APR::Const::ECONNRESET

- since: 2.0.00

Due to possible variants in conditions matching ECONNRESET, for checking error codes against this you most likely want to use the `APR::Status::is_ECONNRESET` function instead.

1.3.2.14 APR::Const::EDSOOPEN

- since: 2.0.00

1.3.2.15 APR::Const::EEXIST

- since: 2.0.00

1.3.2.16 APR::Const::EFTYPE

- since: 2.0.00

1.3.2.17 APR::Const::EGENERAL

- since: 2.0.00

1.3.2.18 APR::Const::EHOSTUNREACH

- since: 2.0.00

1.3.2.19 APR::Const::EINCOMPLETE

- since: 2.0.00

1.3.2.20 APR::Const::EINIT

- since: 2.0.00

1.3.2.21 APR::Const::EINPROGRESS

- since: 2.0.00

1.3.2.22 APR::Const::EINTR

- since: 2.0.00

1.3.2.23 APR::Const::EINVAL

- since: 2.0.00

1.3.2.24 APR::Const::EINVALSOCK

- since: 2.0.00

1.3.2.25 APR::Const::EMFILE

- since: 2.0.00

1.3.2.26 APR::Const::EMISMATCH

- since: 2.0.00

1.3.2.27 APR::Const::ENAMETOOLONG

- since: 2.0.00

1.3.2.28 APR::Const::END

- since: 2.0.00

1.3.2.29 APR::Const::ENETUNREACH

- since: 2.0.00

1.3.2.30 APR::Const::ENFILE

- since: 2.0.00

1.3.2.31 APR::Const::ENODIR

- since: 2.0.00

1.3.2.32 APR::Const::ENOENT

- since: 2.0.00

Due to possible variants in conditions matching ENOENT, for checking error codes against this you most likely want to use the `APR::Status::is_ENOENT` function instead.

1.3.2.33 APR::Const::ENOLOCK

- since: 2.0.00

1.3.2.34 APR::Const::ENOMEM

- since: 2.0.00

1.3.2.35 APR::Const::ENOPOLL

- since: 2.0.00

1.3.2.36 APR::Const::ENOPOOL

- since: 2.0.00

1.3.2.37 APR::Const::ENOPROC

- since: 2.0.00

1.3.2.38 APR::Const::ENOSHMAVAIL

- since: 2.0.00

1.3.2.39 APR::Const::ENOSOCKET

- since: 2.0.00

1.3.2.40 APR::Const::ENOSPC

- since: 2.0.00

1.3.2.41 APR::Const::ENOSTAT

- since: 2.0.00

1.3.2.42 APR::Const::ENOTDIR

- since: 2.0.00

1.3.2.43 APR::Const::ENOTEMPTY

- since: 2.0.00

1.3.2.44 APR::Const::ENOTHDKEY

- since: 2.0.00

1.3.2.45 APR::Const::ENOTHREAD

- since: 2.0.00

1.3.2.46 APR::Const::ENOTIME

- since: 2.0.00

1.3.2.47 APR::Const::ENOTIMPL

Something is not implemented

- since: 2.0.00

1.3.2.48 APR::Const::ENOTSOCK

- since: 2.0.00

1.3.2.49 APR::Const::EOF

- since: 2.0.00

Due to possible variants in conditions matching EOF, for checking error codes against this you most likely want to use the `APR::Status::is_EOF` function instead.

1.3.2.50 APR::Const::EPATHWILD

- since: 2.0.00

1.3.2.51 APR::Const::EPIPE

- since: 2.0.00

1.3.2.52 APR::Const::EPROC_UNKNOWN

- since: 2.0.00

1.3.2.53 APR::Const::ERELATIVE

- since: 2.0.00

1.3.2.54 APR::Const::ESPIPE

- since: 2.0.00

1.3.2.55 APR::Const::ESYMNOTFOUND

- since: 2.0.00

1.3.2.56 APR::Const::ETIMEDOUT

- since: 2.0.00

1.3.2.57 APR::Const::EXDEV

- since: 2.0.00

1.3.3 :fopen

```
use APR::Const -compile => qw(:fopen);
```

The :fopen group is for XXX constants.

1.3.3.1 APR::Const::FOPEN_BINARY

- since: 2.0.00

1.3.3.2 APR::Const::FOPEN_BUFFERED

- since: 2.0.00

1.3.3.3 APR::Const::FOPEN_CREATE

- since: 2.0.00

1.3.3.4 APR::Const::FOPEN_DELONCLOSE

- since: 2.0.00

1.3.3.5 APR::Const::FOPEN_EXCL

- since: 2.0.00

1.3.3.6 APR::Const::FOPEN_PEND

- since: 2.0.00

1.3.3.7 APR::Const::FOPEN_READ

- since: 2.0.00

1.3.3.8 APR::Const::FOPEN_TRUNCATE

- since: 2.0.00

1.3.3.9 APR::Const::FOPEN_WRITE

- since: 2.0.00

1.3.4 :filepath

```
use APR::Const -compile => qw(:filepath);
```

The :filepath group is for XXX constants.

1.3.4.1 APR::Const::FILEPATH_ENCODING_LOCALE

- since: 2.0.00

1.3.4.2 APR::Const::FILEPATH_ENCODING_UNKNOWN

- since: 2.0.00

1.3.4.3 APR::Const::FILEPATH_ENCODING_UTF8

- since: 2.0.00

1.3.4.4 APR::Const::FILEPATH_NATIVE

- since: 2.0.00

1.3.4.5 APR::Const::FILEPATH_NOTABOVEROOT

- since: 2.0.00

1.3.4.6 APR::Const::FILEPATH_NOTABSOLUTE

- since: 2.0.00

1.3.4.7 APR::Const::FILEPATH_NOTRELATIVE

- since: 2.0.00

1.3.4.8 APR::Const::FILEPATH_SECUREROOT

- since: 2.0.00

1.3.4.9 APR::Const::FILEPATH_SECUREROOTTEST

- since: 2.0.00

1.3.4.10 APR::Const::FILEPATH_TRUENAME

- since: 2.0.00

1.3.5 :fprot

```
use APR::Const -compile => qw(:fprot);
```

The :fprot group is used by \$finfo->protection.

1.3.5.1 APR::Const::FPROT_GEXECUTE

Execute by group

- since: 2.0.00

1.3.5.2 APR::Const::FPROT_GREAD

Read by group

- since: 2.0.00

1.3.5.3 APR::Const::FPROT_GSETID

Set group id

- since: 2.0.00

1.3.5.4 APR::Const::FPROT_GWRITE

Write by group

- since: 2.0.00

1.3.5.5 APR::Const::FPROT_OS_DEFAULT

use OS's default permissions

- since: 2.0.00

1.3.5.6 APR::Const::FPROT_UEXECUTE

Execute by user

- since: 2.0.00

1.3.5.7 APR::Const::FPROT_UREAD

Read by user

- since: 2.0.00

1.3.5.8 APR::Const::FPROT_USETID

Set user id

- since: 2.0.00

1.3.5.9 APR::Const::FPROT_UWRITE

Write by user

- since: 2.0.00

1.3.5.10 APR::Const::FPROT_WEXECUTE

Execute by others

- since: 2.0.00

1.3.5.11 APR::Const::FPROT_WREAD

Read by others

- since: 2.0.00

1.3.5.12 APR::Const::FPROT_WSTICKY

Sticky bit

- since: 2.0.00

1.3.5.13 APR::Const::FPROT_WWRITE

Write by others

- since: 2.0.00

1.3.6 :filetype

```
use APR::Const -compile => qw(:filetype);
```

The :filetype group is used by \$finfo->filetype.

1.3.6.1 APR::Const::FILETYPE_BLK

a file is a block device

- since: 2.0.00

1.3.6.2 APR::Const::FILETYPE_CHR

a file is a character device

- since: 2.0.00

1.3.6.3 APR::Const::FILETYPE_DIR

a file is a directory

- since: 2.0.00

1.3.6.4 APR::Const::FILETYPE_LNK

a file is a symbolic link

- since: 2.0.00

1.3.6.5 APR::Const::FILETYPE_NOFILE

the file type is undetermined.

- since: 2.0.00

1.3.6.6 APR::Const::FILETYPE_PIPE

a file is a FIFO or a pipe.

- since: 2.0.00

1.3.6.7 APR::Const::FILETYPE_REG

a file is a regular file.

- since: 2.0.00

1.3.6.8 APR::Const::FILETYPE SOCK

a file is a [unix domain] socket.

- since: 2.0.00

1.3.6.9 APR::Const::FILETYPE_UNKFILE

a file is of some other unknown type or the type cannot be determined.

- since: 2.0.00

1.3.7 :finfo

```
use APR::Const -compile => qw(:finfo);
```

The :finfo group is used by `stat()` and `$finfo->valid`.

1.3.7.1 APR::Const::FINFO_ATIME

Access Time

- since: 2.0.00

1.3.7.2 APR::Const::FINFO_CSIZE

Storage size consumed by the file

- since: 2.0.00

1.3.7.3 APR::Const::FINFO_CTIME

Creation Time

- since: 2.0.00

1.3.7.4 APR::Const::FINFO_DEV

Device

- since: 2.0.00

1.3.7.5 APR::Const::FINFO_DIRENT

an atomic unix apr_dir_read()

- since: 2.0.00

1.3.7.6 APR::Const::FINFO_GPROT

Group protection bits

- since: 2.0.00

1.3.7.7 APR::Const::FINFO_GROUP

Group id

- since: 2.0.00

1.3.7.8 APR::Const::FINFO_ICASE

whether device is case insensitive

- since: 2.0.00

1.3.7.9 APR::Const::FINFO_IDENT

device and inode

- since: 2.0.00

1.3.7.10 APR::Const::FINFO_INODE

Inode

- since: 2.0.00

1.3.7.11 APR::Const::FINFO_LINK

Stat the link not the file itself if it is a link

- since: 2.0.00

1.3.7.12 APR::Const::FINFO_MIN

type, mtime, ctime, atime, size

- since: 2.0.00

1.3.7.13 APR::Const::FINFO_MTIME

Modification Time

- since: 2.0.00

1.3.7.14 APR::Const::FINFO_NAME

name in proper case

- since: 2.0.00

1.3.7.15 APR::Const::FINFO_NLINK

Number of links

- since: 2.0.00

1.3.7.16 APR::Const::FINFO_NORM

All fields provided by an atomic unix apr_stat()

- since: 2.0.00

1.3.7.17 APR::Const::FINFO_OWNER

user and group

- since: 2.0.00

1.3.7.18 APR::Const::FINFO_PROT

all protections

- since: 2.0.00

1.3.7.19 APR::Const::FINFO_SIZE

Size of the file

- since: 2.0.00

1.3.7.20 APR::Const::FINFO_TYPE

Type

- since: 2.0.00

1.3.7.21 APR::Const::FINFO_UPROT

User protection bits

- since: 2.0.00

1.3.7.22 APR::Const::FINFO_USER

User id

- since: 2.0.00

1.3.7.23 APR::Const::FINFO_WPROT

World protection bits

- since: 2.0.00

1.3.8 :flock

```
use APR::Const -compile => qw(:flock);
```

The :flock group is for XXX constants.

1.3.8.1 APR::Const::FLOCK_EXCLUSIVE

- since: 2.0.00

1.3.8.2 APR::Const::FLOCK_NONBLOCK

- since: 2.0.00

1.3.8.3 APR::Const::FLOCK_SHARED

- since: 2.0.00

1.3.8.4 APR::Const::FLOCK_TPEMASK

- since: 2.0.00

1.3.9 :hook

```
use APR::Const -compile => qw(:hook);
```

The :hook group is for XXX constants.

1.3.9.1 APR::Const::HOOK_FIRST

- since: 2.0.00

1.3.9.2 APR::Const::HOOK_LAST

- since: 2.0.00

1.3.9.3 APR::Const::HOOK_MIDDLE

- since: 2.0.00

1.3.9.4 APR::Const::HOOK_REALLY_FIRST

- since: 2.0.00

1.3.9.5 APR::Const::HOOK_REALLY_LAST

- since: 2.0.00

1.3.10 :limit

```
use APR::Const -compile => qw(:limit);
```

The :limit group is for XXX constants.

1.3.10.1 APR::Const::LIMIT_CPU

- since: 2.0.00

1.3.10.2 APR::Const::LIMIT_MEM

- since: 2.0.00

1.3.10.3 APR::Const::LIMIT_NOFILE

- since: 2.0.00

1.3.10.4 APR::Const::LIMIT_NPROC

- since: 2.0.00

1.3.11 :lockmech

```
use APR::Const -compile => qw(:lockmech);
```

The :lockmech group is for XXX constants.

1.3.11.1 APR::Const::LOCK_DEFAULT

- since: 2.0.00

1.3.11.2 APR::Const::LOCK_FCNTL

- since: 2.0.00

1.3.11.3 APR::Const::LOCK_FLOCK

- since: 2.0.00

1.3.11.4 APR::Const::LOCK_POSIXSEM

- since: 2.0.00

1.3.11.5 APR::Const::LOCK_PROC_PTHREAD

- since: 2.0.00

1.3.11.6 APR::Const::LOCK_SYSVSEM

- since: 2.0.00

1.3.12 :poll

```
use APR::Const -compile => qw(:poll);
```

The :poll group is used by poll.

1.3.12.1 APR::Const::POLLERR

- since: 2.0.00

Pending error

1.3.12.2 APR::Const::POLLHUP

- since: 2.0.00

Hangup occurred

1.3.12.3 APR::Const::POLLIN

- **since: 2.0.00**

Can read without blocking

1.3.12.4 APR::Const::POLLNVAL

- **since: 2.0.00**

Descriptor invalid

1.3.12.5 APR::Const::POLLOUT

- **since: 2.0.00**

Can write without blocking

1.3.12.6 APR::Const::POLLPRI

- **since: 2.0.00**

Priority data available

1.3.13 :read_type

```
use APR::Const -compile => qw(:read_type);
```

The `:read_type` group is for IO constants.

1.3.13.1 APR::Const::BLOCK_READ

- **since: 2.0.00**

the read function blocks

1.3.13.2 APR::Const::NONBLOCK_READ

- **since: 2.0.00**

the read function does not block

1.3.14 :shutdown_how

```
use APR::Const -compile => qw(:shutdown_how);
```

The `:shutdown_how` group is for XXX constants.

1.3.14.1 APR::Const::SHUTDOWN_READ

- since: 2.0.00

1.3.14.2 APR::Const::SHUTDOWN_READWRITE

- since: 2.0.00

1.3.14.3 APR::Const::SHUTDOWN_WRITE

- since: 2.0.00

1.3.15 :socket

```
use APR::Const -compile => qw(:socket);
```

The `:socket` group is for the `APR::Socket` object constants, in methods `opt_get` and `opt_set`.

The following section discusses in detail each of the `:socket` constants.

1.3.15.1 APR::Const::SO_DEBUG

Possible values:

XXX

- since: 2.0.00

Turns on debugging information

1.3.15.2 APR::Const::SO_DISCONNECTED

Queries the disconnected state of the socket. (Currently only used on Windows)

Possible values:

XXX

- since: 2.0.00

1.3.15.3 APR::Const::SO_KEEPALIVE

Keeps connections active

Possible values:

XXX

- **since: 2.0.00**

1.3.15.4 APR::Const::SO_LINGER

Lingers on close if data is present

- **since: 2.0.00**

1.3.15.5 APR::Const::SO_NONBLOCK

Turns blocking IO mode on/off for socket.

Possible values:

```
1 nonblocking
0 blocking
```

For example, to set a socket to a blocking IO mode:

```
use APR::Socket ();
use APR::Const    -compile => qw(SO_NONBLOCK);
...
if ($socket->opt_get(APR::Const::SO_NONBLOCK)) {
    $socket->opt_set(APR::Const::SO_NONBLOCK => 0);
}
```

You don't have to query for this option, before setting it. It was done for the demonstration purpose.

- **since: 2.0.00**

1.3.15.6 APR::Const::SO_RCVBUF

Controls the ReceiveBufferSize setting

Possible values:

XXX

- **since: 2.0.00**

1.3.15.7 APR::Const::SO_REUSEADDR

The rules used in validating addresses supplied to bind should allow reuse of local addresses.

Possible values:

XXX

- **since: 2.0.00**

1.3.15.8 `APR::Const::SO_SNDBUF`

Controls the `SendBufferSize` setting

Possible values:

XXX

- **since: 2.0.00**

1.3.16 :status

```
use APR::Const -compile => qw(:status);
```

The `:status` group is for the API that return status code, or set the error variable `XXXXXX`.

The following section discusses in detail each of the available `:status` constants.

1.3.16.1 `APR::Const::TIMEUP`

The operation did not finish before the timeout.

- **since: 2.0.00**

Due to possible variants in conditions matching `TIMEUP`, for checking error codes against this you most likely want to use the `APR::Status::is_TIMEUP` function instead.

1.3.17 :table

```
use APR::Const -compile => qw(:table);
```

The `:table` group is for `overlap()` and `compress()` constants. See `APR::Table` for details.

1.3.17.1 `APR::Const::OVERLAP_TABLES_MERGE`

- **since: 2.0.00**

See `APR::Table::compress` and `APR::Table::overlap`.

1.3.17.2 `APR::Const::OVERLAP_TABLES_SET`

- **since: 2.0.00**

See `APR::Table::compress` and `APR::Table::overlap`.

1.3.18 :uri

```
use APR::Const -compile => qw(:uri);
```

The `:uri` group of constants is for manipulating URIs.

1.3.18.1 APR::Const::URI_ACAP_DEFAULT_PORT

- since: 2.0.00

1.3.18.2 APR::Const::URI_FTP_DEFAULT_PORT

- since: 2.0.00

1.3.18.3 APR::Const::URI_GOPHER_DEFAULT_PORT

- since: 2.0.00

1.3.18.4 APR::Const::URI_HTTPS_DEFAULT_PORT

- since: 2.0.00

1.3.18.5 APR::Const::URI_HTTP_DEFAULT_PORT

- since: 2.0.00

1.3.18.6 APR::Const::URI_IMAP_DEFAULT_PORT

- since: 2.0.00

1.3.18.7 APR::Const::URI_LDAP_DEFAULT_PORT

- since: 2.0.00

1.3.18.8 APR::Const::URI_NFS_DEFAULT_PORT

- since: 2.0.00

1.3.18.9 APR::Const::URI_NNTP_DEFAULT_PORT

- since: 2.0.00

1.3.18.10 APR::Const::URI_POP_DEFAULT_PORT

- since: 2.0.00

1.3.18.11 APR::Const::URI_PROSPERO_DEFAULT_PORT

- since: 2.0.00

1.3.18.12 APR::Const::URI_RTSP_DEFAULT_PORT

- since: 2.0.00

1.3.18.13 APR::Const::URI_SIP_DEFAULT_PORT

- since: 2.0.00

1.3.18.14 APR::Const::URI_SNEWS_DEFAULT_PORT

- since: 2.0.00

1.3.18.15 APR::Const::URI_SSH_DEFAULT_PORT

- since: 2.0.00

1.3.18.16 APR::Const::URI_TELNET_DEFAULT_PORT

- since: 2.0.00

1.3.18.17 APR::Const::URI_TIP_DEFAULT_PORT

- since: 2.0.00

1.3.18.18 APR::Const::URI_UNP_OMITPASSWORD

- since: 2.0.00

See APR::URI::unparse.

1.3.18.19 APR::Const::URI_UNP_OMITPATHINFO

- since: 2.0.00

See APR::URI::unparse.

1.3.18.20 APR::Const::URI_UNP_OMITQUERY

- since: 2.0.00

See APR::URI::unparse.

1.3.18.21 APR::Const::URI_UNP_OMITSITEPART

- since: 2.0.00

See APR::URI::unparse.

1.3.18.22 APR::Const::URI_UNP_OMITUSER

- since: 2.0.00

See APR::URI::unparse.

1.3.18.23 APR::Const::URI_UNP_OMITUSERINFO

- since: 2.0.00

1.3.18.24 APR::Const::URI_UNP_REVEALPASSWORD

- since: 2.0.00

See APR::URI::unparse.

1.3.18.25 APR::Const::URI_WAIS_DEFAULT_PORT

- since: 2.0.00

1.3.19 Other Constants**1.3.19.1 APR::PerlIO::PERLIO_LAYERS_ARE_ENABLED**

- since: 2.0.00

See APR::PerlIO::Constants)

1.4 See Also

mod_perl 2.0 documentation.

1.5 Copyright

mod_perl 2.0 and its core modules are copyrighted under The Apache Software License, Version 2.0.

1.6 Authors

The mod_perl development team and numerous contributors.

Table of Contents:

1	APR::Const - Perl Interface for APR Constants	1
1.1	Synopsis	2
1.2	Description	2
1.3	Constants	2
1.3.1	:common	2
1.3.1.1	APR::Const::SUCCESS	2
1.3.2	:error	2
1.3.2.1	APR::Const::EABOVEROOT	2
1.3.2.2	APR::Const::EABSOLUTE	2
1.3.2.3	APR::Const::EACCES	3
1.3.2.4	APR::Const::EAGAIN	3
1.3.2.5	APR::Const::EBADDATE	3
1.3.2.6	APR::Const::EBADF	3
1.3.2.7	APR::Const::EBADIP	3
1.3.2.8	APR::Const::EBADMASK	3
1.3.2.9	APR::Const::EBADPATH	3
1.3.2.10	APR::Const::EBUSY	3
1.3.2.11	APR::Const::ECONNABORTED	3
1.3.2.12	APR::Const::ECONNREFUSED	4
1.3.2.13	APR::Const::ECONNRESET	4
1.3.2.14	APR::Const::EDSOOPEN	4
1.3.2.15	APR::Const::EEXIST	4
1.3.2.16	APR::Const::EFTYPE	4
1.3.2.17	APR::Const::EGENERAL	4
1.3.2.18	APR::Const::EHOSTUNREACH	4
1.3.2.19	APR::Const::EINCOMPLETE	4
1.3.2.20	APR::Const::EINIT	4
1.3.2.21	APR::Const::EINPROGRESS	4
1.3.2.22	APR::Const::EINTR	5
1.3.2.23	APR::Const::EINVAL	5
1.3.2.24	APR::Const::EINVALSOCK	5
1.3.2.25	APR::Const::EMFILE	5
1.3.2.26	APR::Const::EMISMATCH	5
1.3.2.27	APR::Const::ENAMETOOLONG	5
1.3.2.28	APR::Const::END	5
1.3.2.29	APR::Const::ENETUNREACH	5
1.3.2.30	APR::Const::ENFILE	5
1.3.2.31	APR::Const::ENODIR	5
1.3.2.32	APR::Const::ENOENT	5
1.3.2.33	APR::Const::ENOLOCK	6
1.3.2.34	APR::Const::ENOMEM	6
1.3.2.35	APR::Const::ENOPOLL	6
1.3.2.36	APR::Const::ENOPOOL	6
1.3.2.37	APR::Const::ENOPROC	6

Table of Contents:

1.3.2.38	APR::Const::ENOSHMAVAIL	.6
1.3.2.39	APR::Const::ENOSOCKET	.6
1.3.2.40	APR::Const::ENOSPC	.6
1.3.2.41	APR::Const::ENOSTAT	.6
1.3.2.42	APR::Const::ENOTDIR	.6
1.3.2.43	APR::Const::ENOTEMPTY	.6
1.3.2.44	APR::Const::ENOTHDKEY	.7
1.3.2.45	APR::Const::ENOTHREAD	.7
1.3.2.46	APR::Const::ENOTIME	.7
1.3.2.47	APR::Const::ENOTIMPL	.7
1.3.2.48	APR::Const::ENOTSOCK	.7
1.3.2.49	APR::Const::EOF	.7
1.3.2.50	APR::Const::EPATHWILD	.7
1.3.2.51	APR::Const::EPIPE	.7
1.3.2.52	APR::Const::EPROC_UNKNOWN	.7
1.3.2.53	APR::Const::ERELATIVE	.7
1.3.2.54	APR::Const::ESPIPE	.8
1.3.2.55	APR::Const::ESYMNOTFOUND	.8
1.3.2.56	APR::Const::ETIMEDOUT	.8
1.3.2.57	APR::Const::EXDEV	.8
1.3.3	:fopen	.8
1.3.3.1	APR::Const::FOPEN_BINARY	.8
1.3.3.2	APR::Const::FOPEN_BUFFERED	.8
1.3.3.3	APR::Const::FOPEN_CREATE	.8
1.3.3.4	APR::Const::FOPEN_DELONCLOSE	.8
1.3.3.5	APR::Const::FOPEN_EXCL	.8
1.3.3.6	APR::Const::FOPEN_PEND	.8
1.3.3.7	APR::Const::FOPEN_READ	.9
1.3.3.8	APR::Const::FOPEN_TRUNCATE	.9
1.3.3.9	APR::Const::FOPEN_WRITE	.9
1.3.4	:filepath	.9
1.3.4.1	APR::Const::FILEPATH_ENCODING_LOCALE	.9
1.3.4.2	APR::Const::FILEPATH_ENCODING_UNKNOWN	.9
1.3.4.3	APR::Const::FILEPATH_ENCODING_UTF8	.9
1.3.4.4	APR::Const::FILEPATH_NATIVE	.9
1.3.4.5	APR::Const::FILEPATH_NOTABOVEROOT	.9
1.3.4.6	APR::Const::FILEPATH_NOTABSOLUTE	.9
1.3.4.7	APR::Const::FILEPATH_NOTRELATIVE	.9
1.3.4.8	APR::Const::FILEPATH_SECUREROOT	10
1.3.4.9	APR::Const::FILEPATH_SECUREROOTTEST	10
1.3.4.10	APR::Const::FILEPATH_TRUENAME	10
1.3.5	:fprot	10
1.3.5.1	APR::Const::FPROT_GEXECUTE	10
1.3.5.2	APR::Const::FPROT_GREAD	10
1.3.5.3	APR::Const::FPROT_GSETID	10
1.3.5.4	APR::Const::FPROT_GWRITE	10
1.3.5.5	APR::Const::FPROT_OS_DEFAULT	10

1.3.5.6	APR::Const::FPROT_UEXECUTE	11
1.3.5.7	APR::Const::FPROT_UREAD	11
1.3.5.8	APR::Const::FPROT_USETID	11
1.3.5.9	APR::Const::FPROT_UWRITE	11
1.3.5.10	APR::Const::FPROT_WEXECUTE	11
1.3.5.11	APR::Const::FPROT_WREAD	11
1.3.5.12	APR::Const::FPROT_WSTICKY	11
1.3.5.13	APR::Const::FPROT_WWRITE	12
1.3.6	:filetype	12
1.3.6.1	APR::Const::FILETYPE_BLK	12
1.3.6.2	APR::Const::FILETYPE_CHR	12
1.3.6.3	APR::Const::FILETYPE_DIR	12
1.3.6.4	APR::Const::FILETYPE_LNK	12
1.3.6.5	APR::Const::FILETYPE_NOFILE	12
1.3.6.6	APR::Const::FILETYPE_PIPE	12
1.3.6.7	APR::Const::FILETYPE_REG	13
1.3.6.8	APR::Const::FILETYPE_SOCK	13
1.3.6.9	APR::Const::FILETYPE_UNKFILE	13
1.3.7	:finfo	13
1.3.7.1	APR::Const::FINFO_ETIME	13
1.3.7.2	APR::Const::FINFO_CSIZE	13
1.3.7.3	APR::Const::FINFO_CTIME	13
1.3.7.4	APR::Const::FINFO_DEV	13
1.3.7.5	APR::Const::FINFO_DIRENT	14
1.3.7.6	APR::Const::FINFO_GPROT	14
1.3.7.7	APR::Const::FINFO_GROUP	14
1.3.7.8	APR::Const::FINFO_ICASE	14
1.3.7.9	APR::Const::FINFO_IDENT	14
1.3.7.10	APR::Const::FINFO_INODE	14
1.3.7.11	APR::Const::FINFO_LINK	14
1.3.7.12	APR::Const::FINFO_MIN	14
1.3.7.13	APR::Const::FINFO_MTIME	15
1.3.7.14	APR::Const::FINFO_NAME	15
1.3.7.15	APR::Const::FINFO_NLINK	15
1.3.7.16	APR::Const::FINFO_NORM	15
1.3.7.17	APR::Const::FINFO_OWNER	15
1.3.7.18	APR::Const::FINFO_PROT	15
1.3.7.19	APR::Const::FINFO_SIZE	15
1.3.7.20	APR::Const::FINFO_TYPE	15
1.3.7.21	APR::Const::FINFO_UPROT	16
1.3.7.22	APR::Const::FINFO_USER	16
1.3.7.23	APR::Const::FINFO_WPROT	16
1.3.8	:flock	16
1.3.8.1	APR::Const::FLOCK_EXCLUSIVE	16
1.3.8.2	APR::Const::FLOCK_NONBLOCK	16
1.3.8.3	APR::Const::FLOCK_SHARED	16
1.3.8.4	APR::Const::FLOCK_TYEMASK	16

Table of Contents:

1.3.9	:hook	16
1.3.9.1	APR::Const::HOOK_FIRST	17
1.3.9.2	APR::Const::HOOK_LAST	17
1.3.9.3	APR::Const::HOOK_MIDDLE	17
1.3.9.4	APR::Const::HOOK_REALLY_FIRST	17
1.3.9.5	APR::Const::HOOK_REALLY_LAST	17
1.3.10	:limit	17
1.3.10.1	APR::Const::LIMIT_CPU	17
1.3.10.2	APR::Const::LIMIT_MEM	17
1.3.10.3	APR::Const::LIMIT_NOFILE	17
1.3.10.4	APR::Const::LIMIT_NPROC	17
1.3.11	:lockmech	17
1.3.11.1	APR::Const::LOCK_DEFAULT	18
1.3.11.2	APR::Const::LOCK_FCNTL	18
1.3.11.3	APR::Const::LOCK_FLOCK	18
1.3.11.4	APR::Const::LOCK_POSIXSEM	18
1.3.11.5	APR::Const::LOCK_PROC_PTHREAD	18
1.3.11.6	APR::Const::LOCK_SYSVSEM	18
1.3.12	:poll	18
1.3.12.1	APR::Const::POLLERR	18
1.3.12.2	APR::Const::POLLHUP	18
1.3.12.3	APR::Const::POLLIN	19
1.3.12.4	APR::Const::POLLNVAL	19
1.3.12.5	APR::Const::POLLOUT	19
1.3.12.6	APR::Const::POLLPRI	19
1.3.13	:read_type	19
1.3.13.1	APR::Const::BLOCK_READ	19
1.3.13.2	APR::Const::NONBLOCK_READ	19
1.3.14	:shutdown_how	19
1.3.14.1	APR::Const::SHUTDOWN_READ	20
1.3.14.2	APR::Const::SHUTDOWN_READWRITE	20
1.3.14.3	APR::Const::SHUTDOWN_WRITE	20
1.3.15	:socket	20
1.3.15.1	APR::Const::SO_DEBUG	20
1.3.15.2	APR::Const::SO_DISCONNECTED	20
1.3.15.3	APR::Const::SO_KEEPALIVE	20
1.3.15.4	APR::Const::SO_LINGER	21
1.3.15.5	APR::Const::SO_NONBLOCK	21
1.3.15.6	APR::Const::SO_RCVBUF	21
1.3.15.7	APR::Const::SO_REUSEADDR	21
1.3.15.8	APR::Const::SO_SNDBUF	22
1.3.16	:status	22
1.3.16.1	APR::Const::TIMEUP	22
1.3.17	:table	22
1.3.17.1	APR::Const::OVERLAP_TABLES_MERGE	22
1.3.17.2	APR::Const::OVERLAP_TABLES_SET	22
1.3.18	:uri	23

1.3.18.1	APR::Const::URI_ACAP_DEFAULT_PORT	23
1.3.18.2	APR::Const::URI_FTP_DEFAULT_PORT	23
1.3.18.3	APR::Const::URI_GOPHER_DEFAULT_PORT	23
1.3.18.4	APR::Const::URI_HTTPS_DEFAULT_PORT	23
1.3.18.5	APR::Const::URI_HTTP_DEFAULT_PORT	23
1.3.18.6	APR::Const::URI_IMAP_DEFAULT_PORT	23
1.3.18.7	APR::Const::URI_LDAP_DEFAULT_PORT	23
1.3.18.8	APR::Const::URI_NFS_DEFAULT_PORT	23
1.3.18.9	APR::Const::URI_NNTP_DEFAULT_PORT	23
1.3.18.10	APR::Const::URI_POP_DEFAULT_PORT	24
1.3.18.11	APR::Const::URI_PROSPERO_DEFAULT_PORT	24
1.3.18.12	APR::Const::URI_RTSP_DEFAULT_PORT	24
1.3.18.13	APR::Const::URI_SIP_DEFAULT_PORT	24
1.3.18.14	APR::Const::URI_SNEWS_DEFAULT_PORT	24
1.3.18.15	APR::Const::URI_SSH_DEFAULT_PORT	24
1.3.18.16	APR::Const::URI_TELNET_DEFAULT_PORT	24
1.3.18.17	APR::Const::URI_TIP_DEFAULT_PORT	24
1.3.18.18	APR::Const::URI_UNP_OMITPASSWORD	24
1.3.18.19	APR::Const::URI_UNP_OMITPATHINFO	24
1.3.18.20	APR::Const::URI_UNP_OMITQUERY	25
1.3.18.21	APR::Const::URI_UNP_OMITSITEPART	25
1.3.18.22	APR::Const::URI_UNP_OMITUSER	25
1.3.18.23	APR::Const::URI_UNP_OMITUSERINFO	25
1.3.18.24	APR::Const::URI_UNP_REVEALPASSWORD	25
1.3.18.25	APR::Const::URI_WAIS_DEFAULT_PORT	25
1.3.19	Other Constants	25
1.3.19.1	APR::Perlio::PERLIO_LAYERS_ARE_ENABLED	25
1.4	See Also	25
1.5	Copyright	26
1.6	Authors	26