

NAME

perl5211delta - what is new for perl v5.21.1

DESCRIPTION

This document describes differences between the 5.21.0 release and the 5.21.1 release.

If you are upgrading from an earlier release such as 5.20.0, first read *perl5210delta*, which describes differences between 5.20.0 and 5.21.0.

Notice

This release removes a number of previously deprecated constructs, many that have been around for a long time. Please see *Incompatible Changes* for more information.

Core Enhancements

Unicode 7.0 is now supported

For details on what is in this release, see <http://www.unicode.org/versions/Unicode7.0.0/>.

Experimental C Backtrace API

Starting from Perl 5.21.1, on some platforms Perl supports retrieving the C level backtrace (similar to what symbolic debuggers like gdb do).

The backtrace returns the stack trace of the C call frames, with the symbol names (function names), the object names (like "perl"), and if it can, also the source code locations (file:line).

The supported platforms are Linux and OS X (some *BSD might work at least partly, but they have not yet been tested).

The feature needs to be enabled with `Configure -Dusebacktrace`.

Also included is a C API to retrieve backtraces.

See "*C backtrace*" in *perlhacktips* for more information.

qr/foo/x now ignores any Unicode pattern white space

The `/x` regular expression modifier allows the pattern to contain white space and comments, both of which are ignored, for improved readability. Until now, not all the white space characters that Unicode designates for this purpose were handled. The additional ones now recognized are U+0085 NEXT LINE, U+200E LEFT-TO-RIGHT MARK, U+200F RIGHT-TO-LEFT MARK, U+2028 LINE SEPARATOR, and U+2029 PARAGRAPH SEPARATOR.

use locale can restrict which locale categories are affected

It is now possible to pass a parameter to `use locale` to specify a subset of locale categories to be locale-aware, with the remaining ones unaffected. See "*The 'use locale' pragma*" in *perllocale* for details.

Incompatible Changes

\N{ with a sequence of multiple spaces is now a fatal error.

This has been deprecated since v5.18.

In double-quotish \cX, X must now be a printable ASCII character

In prior releases, failure to do this raised a deprecation warning.

Splitting the tokens (? and (* in regular expressions is now a fatal compilation error.

These had been deprecated since v5.18.

5 additional characters are treated as white space under /x in regex patterns (unless escaped)

The use of these characters with /x outside bracketed character classes and when not preceded by a backslash has raised a deprecation warning since v5.18. Now they will be ignored. See *qr/foo/x* for the list of the five characters.

Comment lines within (?[]) now are ended only by a \n

(?[]) is an experimental feature, introduced in v5.18. It operates as if /x is always enabled. But there was a difference, comment lines (following a # character) were terminated by anything matching \R which includes all vertical whitespace, such as form feeds. For consistency, this is now changed to match what terminates comment lines outside (?[]), namely a \n (even if escaped), which is the same as what terminates a heredoc string and formats.

Omitting % and @ on hash and array names is no longer permitted

Really old Perl let you omit the @ on array names and the % on hash names in some spots. This has issued a deprecation warning since Perl 5.0, and is no longer permitted.

"\$!" text is now in English outside "use locale" scope

Previously, the text, unlike almost everything else, always came out based on the current underlying locale of the program. (Also affected on some systems is "\$^E".) For programs that are unprepared to handle locale, this can cause garbage text to be displayed. It's better to display text that is translatable via some tool than garbage text which is much harder to figure out.

"\$!" text will be returned in UTF-8 when appropriate

The stringification of \$! and \$^E will have the UTF-8 flag set when the text is actually non-ASCII UTF-8. This will enable programs that are set up to be locale-aware to properly output messages in the user's native language. Code that needs to continue the 5.20 and earlier behavior can do the stringification within the scopes of both 'use bytes' and 'use locale ":messages"'. No other Perl operations will be affected by locale; only \$! and \$^E stringification. The 'bytes' pragma causes the UTF-8 flag to not be set, just as in previous Perl releases. This resolves [perl #112208].

Support for ?PATTERN? without explicit operator has been removed

Starting regular expressions matching only once directly with the question mark delimiter is now a syntax error, so that the question mark can be available for use in new operators. Write m?PATTERN? instead, explicitly using the m operator: the question mark delimiter still invokes match-once behaviour.

defined(@array) and defined(%hash) are now fatal errors

These have been deprecated since v5.6.1 and have raised deprecation warnings since v5.16.

Using a hash or an array as a reference are now fatal errors.

For example, %foo->{"bar"} now causes a fatal compilation error. These have been deprecated since before v5.8, and have raised deprecation warnings since then.

Deprecations

Using a NO-BREAK space in a character alias for \N{...} is now deprecated

This non-graphic character is essentially indistinguishable from a regular space, and so should not be allowed. See *"CUSTOM ALIASES" in charnames*.

A literal "{" should now be escaped in a pattern

If you want a literal left curly bracket (also called a left brace) in a regular expression pattern, you should now escape it by either preceding it with a backslash ("\\{") or enclosing it within square brackets "[{]", or by using \Q; otherwise a deprecation warning will be raised. This was first announced as forthcoming in the v5.16 release; it will allow future extensions to the language to happen.

Performance Enhancements

- Many internal functions have been refactored to improve performance and reduce their memory footprints.
[perl #121436] [perl #121906] [perl #121969]
- `-T` and `-B` filetests will return sooner when an empty file is detected.
perl #121489

Modules and Pragmata

Updated Modules and Pragmata

- The libnet collection of modules has been upgraded from version 1.25 to 1.27.
There are only whitespace changes to the installed files.
- A mismatch between the documentation and the code in `utf8::downgrade()` was fixed in favour of the documentation. The optional second argument is now correctly treated as a perl boolean (true/false semantics) and not as an integer.
- The Locale-Codes collection of modules has been upgraded from version 3.30 to 3.31.
Fixed a bug in the scripts used to extract data from spreadsheets that prevented the SHP currency code from being found. *[cpan #94229]*
- *Archive::Tar* has been upgraded from version 1.96 to 2.00.
- *autodie* has been upgraded from version 2.23 to 2.25.
- *B* has been upgraded from version 1.48 to 1.49.
- *B::Deparse* has been upgraded from version 1.26 to 1.27.
- *Benchmark* has been upgraded from version 1.18 to 1.19.
- *Carp* has been upgraded from version 1.3301 to 1.34.
Carp::Heavy now ignores version mismatches with *Carp* if *Carp* is newer than 1.12, since *Carp::Heavy*'s guts were merged into *Carp* at that point. *[perl #121574]*
- *charnames* has been upgraded from version 1.40 to 1.41.
- *CPAN::Meta* has been upgraded from version 2.140640 to 2.141520.
- *Data::Dumper* has been upgraded from version 2.151 to 2.152.
Changes to resolve Coverity issues.
XS dumps incorrectly stored the name of code references stored in a GLOB. *[perl #122070]*
- *Devel::Peek* has been upgraded from version 1.16 to 1.17.
- *Devel::PPPort* has been upgraded from version 3.21 to 3.24.
- *Digest::SHA* has been upgraded from version 5.88 to 5.92.
- *DynaLoader* has been upgraded from version 1.25 to 1.26.
- *Encode* has been upgraded from version 2.60 to 2.62.
piconv now has better error handling when the encoding name is nonexistent, and a build breakage when upgrading *Encode* in perl-5.8.2 and earlier has been fixed.
- *Errno* has been upgraded from version 1.20_03 to 1.20_04.
- *Exporter* has been upgraded from version 5.70 to 5.71.

- *ExtUtils::Install* has been upgraded from version 1.67 to 1.68.
- *ExtUtils::Miniperl* has been upgraded from version 1.01 to 1.02.
- *ExtUtils::ParseXS* has been upgraded from version 3.24 to 3.25.
- *ExtUtils::Typemaps* has been upgraded from version 3.24 to 3.25.
- *File::Spec* has been upgraded from version 3.47 to 3.48.
- *Hash::Util* has been upgraded from version 0.16 to 0.17.
Minor bug fixes and documentation fixes to *Hash::Util::hash_stats()*
- *IO* has been upgraded from version 1.31 to 1.32.
- *List::Util* has been upgraded from version 1.38 to 1.39.
- *locale* has been upgraded from version 1.03 to 1.04.
- *Locale::Codes* has been upgraded from version 3.30 to 3.31.
- *Math::BigInt* has been upgraded from version 1.9993 to 1.9995.
Synchronize POD changes from the CPAN release.
Math::BigInt->blog(x) would sometimes return *blog(2*x)* when the accuracy was greater than 70 digits.
The result of *Math::BigInt->bdiv()* in list context now satisfies *x = quotient * divisor + remainder*.
- *Math::BigRat* has been upgraded from version 0.2606 to 0.2608.
Synchronize POD changes from the CPAN release.
- *Module::CoreList* has been upgraded from version 5.021001 to 5.021001_01.
- *Module::Metadata* has been upgraded from version 1.000019 to 1.000024.
Support installations on older perls with an *ExtUtils::MakeMaker* earlier than 6.63_03
- *NDBM_File* has been upgraded from version 1.12 to 1.13.
- *OS2::Process* has been upgraded from version 1.09 to 1.10.
- *perl5db.pl* has been upgraded from version 1.44 to 1.45.
fork() in the debugger under *tmux* will now create a new window for the forked process. [*perl* #121333]
The debugger now saves the current working directory on startup and restores it when you restart your program with *R* or *rerun*. [*perl* #121509]
- *PerlIO::encoding* has been upgraded from version 0.18 to 0.19.
No changes in behaviour.
- *PerlIO::mmap* has been upgraded from version 0.011 to 0.013.
No changes in behaviour.
- *PerlIO::scalar* has been upgraded from version 0.18 to 0.19.
No changes in behaviour.
- *PerlIO::via* has been upgraded from version 0.14 to 0.15.
- *Pod::Html* has been upgraded from version 1.21 to 1.22.
- *POSIX* has been upgraded from version 1.38_03 to 1.40.

- *Scalar::Util* has been upgraded from version 1.38 to 1.39.
- *SelfLoader* has been upgraded from version 1.21 to 1.22.
- *Socket* has been upgraded from version 2.013 to 2.014.
- *Storable* has been upgraded from version 2.49 to 2.51.
- *Term::ANSIColor* has been upgraded from version 4.02 to 4.03.
- *Test::Harness* has been upgraded from version 3.30 to 3.32.
- *Test::Simple* has been upgraded from version 1.001002 to 1.001003.
- *threads* has been upgraded from version 1.93 to 1.94.
- *Tie::File* has been upgraded from version 1.00 to 1.01.
- *Unicode::Collate* has been upgraded from version 1.04 to 1.07.
Version 0.67's improved discontinuous contractions is invalidated by default and is supported as a parameter 'long_contraction'.
- *Unicode::Normalize* has been upgraded from version 1.17 to 1.18.
The XSUB implementation has been removed in favour of pure Perl.
- *Unicode::UCD* has been upgraded from version 0.57 to 0.58.
- *utf8* has been upgraded from version 1.13 to 1.14.
- *warnings* has been upgraded from version 1.23 to 1.24.

Documentation

Changes to Existing Documentation

perlfunc

- `-l` now notes that it will return false if symlinks aren't supported by the file system.
[perl #121523]
- Note that `exec LIST` and `system LIST` may fall back to the shell on Win32. Only `exec PROGRAM LIST` and `system PROGRAM LIST` indirect object syntax will reliably avoid using the shell.
This has also been noted in *perlport*.
[perl #122046]

perlapi

- Note that `SvSetSV` doesn't do set magic.
- `sv_usepvn_flags` - Fix documentation to mention the use of `NewX` instead of `malloc`.
[perl #121869]
- Clarify where `NUL` may be embedded or is required to terminate a string.
- Previously missing documentation due to formatting errors are now included.
- Entries are now organized into groups rather than by file where they are found.
- Alphabetical sorting of entries is now handled by the POD generator to make entries easier to find when scanning.

perlhacktips

- Updated documentation for the `test.valgrind` make target.
[perl #121431]

perlre

- The `/x` modifier has been clarified to note that comments cannot be continued onto the next line by escaping them.

Unicode::UCD

- The documentation includes many clarifications and fixes.

Diagnostics

The following additions or changes have been made to diagnostic output, including warnings and fatal error messages. For the complete list of diagnostic messages, see *perldiag*.

New Diagnostics

New Errors

- *In '(?...)', the '(' and '?' must be adjacent in regex; marked by <-- HERE in m/%s/*
(F) The two-character sequence "`(?`" in this context in a regular expression pattern should be an indivisible token, with nothing intervening between the "`(`" and the "`?`", but you separated them.
- *In '(*VERB...)', the '(' and '*' must be adjacent in regex; marked by <-- HERE in m/%s/*
(F) The two-character sequence "`(*`" in this context in a regular expression pattern should be an indivisible token, with nothing intervening between the "`(`" and the "`*`", but you separated them.
- *charnames alias definitions may not contain a sequence of multiple spaces*
(F) You defined a character name which had multiple space characters in a row. Change them to single spaces. Usually these names are defined in the `:alias` import argument to `use charnames`, but they could be defined by a translator installed into `$_H{charnames}`. See *"CUSTOM ALIASES" in charnames*.
- *charnames alias definitions may not contain trailing white-space*
(F) You defined a character name which ended in a space character. Remove the trailing space(s). Usually these names are defined in the `:alias` import argument to `use charnames`, but they could be defined by a translator installed into `$_H{charnames}`. See *"CUSTOM ALIASES" in charnames*.
- *Can't use a hash as a reference*
(F) You tried to use a hash as a reference, as in `%foo->{"bar"}` or `$_$ref->{"hello"}`. Versions of perl `<= 5.6.1` used to allow this syntax, but shouldn't have.
- *Can't use an array as a reference*
(F) You tried to use an array as a reference, as in `@foo->[23]` or `$_$ref->[99]`. Versions of perl `<= 5.6.1` used to allow this syntax, but shouldn't have.
- *Can't use 'defined(@array)' (Maybe you should just omit the defined())?*
(F) `defined()` is not useful on arrays because it checks for an undefined scalar value. If you want to see if the array is empty, just use `if (@array) { # not empty }` for example.
- *Can't use 'defined(%hash)' (Maybe you should just omit the defined())?*
(F) `defined()` is not usually right on hashes.
Although `defined %hash` is false on a plain not-yet-used hash, it becomes true in several

non-obvious circumstances, including iterators, weak references, stash names, even remaining true after `undef %hash`. These things make `defined %hash` fairly useless in practice, so it now generates a fatal error.

If a check for non-empty is what you wanted then just put it in boolean context (see "*Scalar values*" in *perldata*):

```
if (%hash) {
    # not empty
}
```

If you had `defined %Foo::Bar::QUUX` to check whether such a package variable exists then that's never really been reliable, and isn't a good way to enquire about the features of a package, or whether it's loaded, etc.

- *Illegal suidscript*

(F) The script run under `suidperl` was somehow illegal.

New Warnings

- *Unescaped left brace in regex is deprecated, passed through in regex; marked by <-- HERE in m/%s/*

(D deprecated, regexp) You used a literal `"{"` character in a regular expression pattern. You should change to use `"\{"` instead, because a future version of Perl (tentatively v5.26) will consider this to be a syntax error. If the pattern delimiters are also braces, any matching right brace (`"}"`) should also be escaped to avoid confusing the parser, for example,

```
qr{abc\{def\}ghi}
```

- *NO-BREAK SPACE in a charnames alias definition is deprecated*

(D deprecated) You defined a character name which contained a no-break space character. Change it to a regular space. Usually these names are defined in the `:alias` import argument to `use charnames`, but they could be defined by a translator installed into `$_H{charnames}`. See "*CUSTOM ALIASES*" in *charnames*.

- *PerlIO layer ':win32' is experimental*

(S experimental::win32_perlio) The `:win32` PerlIO layer is experimental. If you want to take the risk of using this layer, simply disable this warning:

```
no warnings "experimental::win32_perlio";
```

- *Negative repeat count does nothing*

(W numeric) You tried to execute the `x` repetition operator fewer than 0 times, which doesn't make sense.

- *localtime(%f) failed*

(W overflow) You called `localtime` with a number that it could not handle: too large, too small, or NaN. The returned value is `undef`.

- *gmtime(%f) failed*

(W overflow) You called `gmtime` with a number that it could not handle: too large, too small, or NaN. The returned value is `undef`.

- *PerlIO layer ':win32' is experimental:*

(S experimental::win32_perlio) The `:win32` PerlIO layer is experimental. If you want to take the risk of using this layer, simply disable this warning:

```
no warnings "experimental::win32_perlio";
```


- *Negative repeat count does nothing*
(W numeric) This warns when the repeat count of the `x` repetition operator is negative.
This warning may be changed or removed if it turn out that it was unwise to have added it.

Changes to Existing Diagnostics

- *Variable length lookbehind not implemented in regex `m/%s/`*
Information about Unicode behaviour has been added.
- *<> should be quotes*
This warning has been changed to *<> at require-statement should be quotes* to make the issue more identifiable.
- *Unsuccessful %s on filename containing newline*
This warning is now only produced when the newline is at the end of the filename.

Utility Changes

x2p/

- The `x2p/` directory has been removed from the Perl core.
This removes `find2perl`, `s2p` and `a2p`. They have all been released to CPAN as separate distributions (`App::find2perl`, `App::s2p`, `App::a2p`).

Configuration and Compilation

- `make test.valgrind` now supports parallel testing.
For example:

```
TEST_JOBS=9 make test.valgrind
```


See "*valgrind*" in *perlhacktips* for more information.
[perl #121431]
- The MAD (Misc Attribute Decoration) build option has been removed
This was an unmaintained attempt at preserving the Perl parse tree more faithfully so that automatic conversion of Perl 5 to Perl 6 would have been easier.
This build-time configuration option had been unmaintained for years, and had probably seriously diverged on both Perl 5 and Perl 6 sides.

Platform Support

Discontinued Platforms

NeXTSTEP/OPENSTEP

NeXTSTEP was proprietary OS bundled with NeXT's workstations in the early to mid 90's; OPENSTEP was an API specification that provided a NeXTSTEP-like environment on a non-NeXTSTEP system. Both are now long dead, so support for building Perl on them has been removed.

Platform-Specific Notes

OpenBSD

On OpenBSD, Perl will now default to using the system `malloc` due to the security features it provides. Perl's own `malloc` wrapper has been in use since v5.14 due to performance reasons, but the OpenBSD project believes the tradeoff is worth it and would prefer that users who need the speed specifically ask for it.

[perl #122000].

Internal Changes

- The deprecated variable `PL_sv_objcount` has been removed.
- Perl now tries to keep the locale category `LC_NUMERIC` set to "C" except around operations that need it to be set to the program's underlying locale. This protects the many XS modules that cannot cope with the decimal radix character not being a dot. Prior to this release, Perl initialized this category to "C", but a call to `POSIX::setlocale()` would change it. Now such a call will change the underlying locale of the `LC_NUMERIC` category for the program, but the locale exposed to XS code will remain "C". There is an API under development for those relatively few modules that need to use the underlying locale. This API will be nailed down during the course of developing v5.21. Send email to <mailto:perl5-porters@perl.org> for guidance.
- A new macro `isUTF8_CHAR` has been written which efficiently determines if the string given by its parameters begins with a well-formed UTF-8 encoded character.

Selected Bug Fixes

- `index()` and `rindex()` no longer crash when used on strings over 2GB in size. [[perl #121562](#)].
- A small previously intentional memory leak in `PERL_SYS_INIT/PERL_SYS_INIT3` on Win32 builds was fixed. This might affect embedders who repeatedly create and destroy perl engines within the same process.
- `POSIX::localeconv()` now returns the data for the program's underlying locale even when called from outside the scope of `use locale`.
- `POSIX::localeconv()` now works properly on platforms which don't have `LC_NUMERIC` and/or `LC_MONETARY`, or for which Perl has been compiled to disregard either or both of these locale categories. In such circumstances, there are now no entries for the corresponding values in the hash returned by `localeconv()`.
- `POSIX::localeconv()` now marks appropriately the values it returns as UTF-8 or not. Previously they were always returned as a bytes, even if they were supposed to be encoded as UTF-8.
- On Microsoft Windows, within the scope of `use locale`, the following POSIX character classes gave results for many locales that did not conform to the POSIX standard: `[:alnum:]`, `[:alpha:]`, `[:blank:]`, `[:digit:]`, `[:graph:]`, `[:lower:]`, `[:print:]`, `[:punct:]`, `[:upper:]`, `[:word:]`, and `[:xdigit:]`. These are because the underlying Microsoft implementation does not follow the standard. Perl now takes special precautions to correct for this.
- Many issues have been detected by *Coverity* and fixed.
- `system()` and friends should now work properly on more Android builds.
Due to an oversight, the value specified through `-Dtargetsh` to `Configure` would end up being ignored by some of the build process. This caused perls cross-compiled for Android to end up with defective versions of `system()`, `exec()` and `backticks`: the commands would end up looking for `/bin/sh` instead of `/system/bin/sh`, and so would fail for the vast majority of devices, leaving `$!` as `ENOENT`.
- `qr(...\(...\)...)`, `qr[...\[...\]]...`, and `qr{...\[...\]}...` now work. Previously it was impossible to escape these three left-characters with a backslash within a regular expression pattern where otherwise they would be considered metacharacters, and the pattern opening delimiter was the character, and the closing delimiter was its mirror character.

Acknowledgements

Perl 5.21.1 represents approximately 3 weeks of development since Perl 5.21.0 and contains approximately 240,000 lines of changes across 680 files from 37 authors.

Excluding auto-generated files, documentation and release tools, there were approximately 150,000 lines of changes to 420 .pm, .t, .c and .h files.

Perl continues to flourish into its third decade thanks to a vibrant community of users and developers. The following people are known to have contributed the improvements that became Perl 5.21.1:

Alex Solovey, Andrew Fresh, Andy Dougherty, Brian Fraser, Chris 'BinGOs' Williams, Craig A. Berry, Dagfinn Ilmari MannsÅker, Daniel Dragan, Darin McBride, David Mitchell, Doug Bell, H.Merijn Brand, James E Keenan, Jarkko Hietaniemi, kafka, Karen Etheridge, Karl Williamson, Leon Timmermans, Matthew Horsfall, Michael Bunk, Nicholas Clark, Niels Thykier, Norman Koch, Peter John Acklam, Pierre Bogossian, Reini Urban, Ricardo Signes, Rob Hoelz, Shlomi Fish, Smylers, Steffen MÅller, Steve Hay, Sullivan Beck, Thomas Sibley, Todd Rinaldo, Tony Cook, Yves Orton.

The list above is almost certainly incomplete as it is automatically generated from version control history. In particular, it does not include the names of the (very much appreciated) contributors who reported issues to the Perl bug tracker.

Many of the changes included in this version originated in the CPAN modules included in Perl's core. We're grateful to the entire CPAN community for helping Perl to flourish.

For a more complete list of all of Perl's historical contributors, please see the *AUTHORS* file in the Perl source distribution.

Reporting Bugs

If you find what you think is a bug, you might check the articles recently posted to the comp.lang.perl.misc newsgroup and the perl bug database at <https://rt.perl.org/>. There may also be information at <http://www.perl.org/>, the Perl Home Page.

If you believe you have an unreported bug, please run the *perlbug* program included with your release. Be sure to trim your bug down to a tiny but sufficient test case. Your bug report, along with the output of `perl -V`, will be sent off to perlbug@perl.org to be analysed by the Perl porting team.

If the bug you are reporting has security implications, which make it inappropriate to send to a publicly archived mailing list, then please send it to perl5-security-report@perl.org. This points to a closed subscription unarchived mailing list, which includes all the core committers, who will be able to help assess the impact of issues, figure out a resolution, and help co-ordinate the release of patches to mitigate or fix the problem across all platforms on which Perl is supported. Please only use this address for security issues in the Perl core, not for modules independently distributed on CPAN.

SEE ALSO

The *Changes* file for an explanation of how to view exhaustive details on what changed.

The *INSTALL* file for how to build Perl.

The *README* file for general stuff.

The *Artistic* and *Copying* files for copyright information.