

NAME

perl5215delta - what is new for perl v5.21.5

DESCRIPTION

This document describes differences between the 5.21.4 release and the 5.21.5 release.

If you are upgrading from an earlier release such as 5.21.3, first read *perl5214delta*, which describes differences between 5.21.3 and 5.21.4.

Core Enhancements

New double-diamond operator

`<<>>` is like `<>` but uses three-argument `open` to open each file in `@ARGV`. So each element of `@ARGV` is an actual file name, and `"|foo"` won't be treated as a pipe open.

Aliasing via reference

Variables and subroutines can now be aliased by assigning to a reference:

```
\$c = \$d;  
\&x = \&y;
```

Or by using a backslash before a `foreach` iterator variable, which is perhaps the most useful idiom this feature provides:

```
foreach \%hash (@array_of_hash_refs) { ... }
```

This feature is experimental and must be enabled via `use feature 'refaliasing'`. It will warn unless the `experimental::refaliasing` warnings category is disabled.

See *"Assigning to References" in perlref*

Perl now supports POSIX 2008 locale currency additions.

On platforms that are able to handle POSIX.1-2008, the hash returned by `POSIX::localeconv()` includes the international currency fields added by that version of the POSIX standard. These are `int_n_cs_precedes`, `int_n_sep_by_space`, `int_n_sign_posn`, `int_p_cs_precedes`, `int_p_sep_by_space`, and `int_p_sign_posn`.

Packing infinity or not-a-number into a character is now fatal

Before, when trying to pack infinity or not-a-number into a (signed) character, Perl would warn, and assumed you tried to pack `0xFF`; if you gave it as an argument to `chr`, `U+FFFD` was returned.

But now, all such actions (`pack`, `chr`, and `print '%c'`) result in a fatal error.

Inf and NaN

Many small improvements, bug fixes and added test cases for dealing with math related to infinity and not-a-number.

Security

Perl is now compiled with `-fstack-protector-strong` if available

Perl has been compiled with the anti-stack-smashing option `-fstack-protector` since 5.10.1. Now Perl uses the newer variant called `-fstack-protector-strong`, if available. (This was added already in 5.21.4.)

Deprecations

Use of multiple `/x` regexp modifiers

It is now deprecated to say something like any of the following:

```
qr/foo/xx;  
/(?xax:foo)/;  
use re qw(/amxx);
```

That is, now `x` should only occur once in any string of contiguous regular expression pattern modifiers. We do not believe there are any occurrences of this in all of CPAN. This is in preparation for a future Perl release having `/xx` mean to allow white-space for readability in bracketed character classes (those enclosed in square brackets: `[...]`).

Performance Enhancements

- `length` is up to 20% faster for non-magical/non-tied scalars containing a string if it is a non-utf8 string or if `use bytes;` is in scope.
- Non-magical/non-tied scalars that contain only a floating point value and are on most Perl builds with 64 bit integers now use 8-32 less bytes of memory depending on OS.
- In `@array = split`, the assignment can be optimized away with `split` writing directly to the array. This optimisation was happening only for package arrays other than `@_` and only sometimes. Now this optimisation happens almost all the time.
- `join` is now subject to constant folding. Moreover, `join` with a scalar or constant for the separator and a single-item list to join is simplified to a stringification. The separator doesn't even get evaluated.
- `qq(@array)` is implemented using two ops: a stringify op and a join op. If the qq contains nothing but a single array, the stringification is optimized away.
- `our $var` and `our($s,@a,%h)` in void context are no longer evaluated at run time. Even a whole sequence of `our $foo;` statements will simply be skipped over. The same applies to state variables.

Modules and Pragmata

Updated Modules and Pragmata

- *attributes* has been upgraded from version 0.23 to 0.24.
Avoid reading beyond the end of a buffer. [perl #122629]
- *B* has been upgraded from version 1.51 to 1.52.
- *B::Concise* has been upgraded from version 0.993 to 0.994.
Null ops that are part of the execution chain are now given sequence numbers.
Private flags for nulled ops are now dumped with mnemonics as they would be for the non-nulled counterparts.
B::Deparse has been upgraded from version 1.28 to 1.29.
Parenthesised arrays in lists passed to `\` are now correctly deparsed with parentheses (e.g., `\(@a, (@b), @c)` now retains the parentheses around `@b`), this preserving the flattening behaviour of referenced parenthesised arrays. Formerly, it only worked for one array: `\(@a)`.
`local our` is now deparsed correctly, with the `our` included.
`for($foo; !$bar; $baz) {...}` was deparsed without the `!` (or `not`). This has been fixed.
Core keywords that conflict with lexical subroutines are now deparsed with the `CORE::` prefix.
`foreach state $x (...)` now deparses correctly with `state` and not `my`.
`our @array = split(...)` now deparses correctly with `our` in those cases where the assignment is optimized away.
- *B::Debug* has been upgraded from version 1.21 to 1.22.

- *B::Deparse* has been upgraded from version 1.28 to 1.29.
- *Compress::Raw::Bzip2* has been upgraded from version 2.064 to 2.066.
- *Compress::Raw::Zlib* has been upgraded from version 2.065 to 2.066.
- *CPAN::Meta* has been upgraded from version 2.142060 to 2.142690.
- *DynaLoader* has been upgraded from version 1.26 to 1.27.
Remove `dl_nonlazy` global if unused in Dynaloader. [perl #122926]
- *Errno* has been upgraded from version 1.20_04 to 1.21.
- *experimental* has been upgraded from version 0.010 to 0.012.
- *ExtUtils::CBuilder* has been upgraded from version 0.280219 to 0.280220.
- *ExtUtils::Miniperl* has been upgraded from version 1.02 to 1.03.
- *Fcntl* has been upgraded from version 1.11 to 1.13.
Add support for the Linux pipe buffer size `fcntl()` commands.
- *feature* has been upgraded from version 1.37 to 1.38.
- *File::Find* has been upgraded from version 1.28 to 1.29.
Slightly faster module loading time.
- *File::Spec* has been upgraded from version 3.50 to 3.51.
- *HTTP::Tiny* has been upgraded from version 0.049 to 0.050.
- The IO-Compress set of modules has been upgraded from version 2.064 to 2.066.
- *JSON::PP* has been upgraded from version 2.27203 to 2.27300.
- The libnet collection of modules has been upgraded from version 1.27 to 3.02.
Support for IPv6 and SSL to `Net::FTP`, `Net::NNTP`, `Net::POP3` and `Net::SMTP`.
Improvements in `Net::SMTP` authentication.
- *Module::CoreList* has been upgraded from version 5.20140920 to 5.20141020.
Updated to cover the latest releases of Perl.
- *Opcodes* has been upgraded from version 1.28 to 1.29.
- The PathTools module collection has been upgraded from version 3.50 to 3.51.
Slightly faster module loading time.
- *perlfaq* has been upgraded from version 5.0150045 to version 5.0150046. [perl #123008]
- *POSIX* has been upgraded from version 1.43 to 1.45.
`POSIX::tmpnam()` now produces a deprecation warning. [perl #122005]
- *re* has been upgraded from version 0.26 to 0.27.
- *Socket* has been upgraded from version 2.015 to 2.016.
- *Test::Simple* has been upgraded from version 1.001006 to 1.001008.
- *threads::shared* has been upgraded from version 1.46 to 1.47.
- *warnings* has been upgraded from version 1.26 to 1.28.

- *XSLoader* has been upgraded from version 0.17 to 0.18.
Allow XSLoader to load modules from a different namespace. [perl #122455]

Documentation

Changes to Existing Documentation

perlrecharclass

- Clarifications have been added to "*Character Ranges*" in *perlrecharclass* to the effect that Perl guarantees that `[A-Z]`, `[a-z]`, `[0-9]` and any subranges thereof in regular expression bracketed character classes are guaranteed to match exactly what a naive English speaker would expect them to match, even on platforms (such as EBCDIC) where special handling is required to accomplish this.

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

- *Cannot chr %f*
- *Cannot compress %f in pack*
- *Cannot pack %f with '%c'*
- *Cannot print %f with '%c'*

Changes to Existing Diagnostics

- `"my" variable &foo::bar can't be in a package` has been reworded to say `'subroutine'` instead of `'variable'`.

Testing

- Some regular expression tests are written in such a way that they will run very slowly if certain optimizations break. These tests have been moved into new files, *t/re/speed.t* and *t/re/speed_thr.t*, and are run with a `watchdog()`.

Platform Support

Regained Platforms

IRIX and Tru64 platforms are working again. (Some `make test` failures remain.)

Platform-Specific Notes

EBCDIC

Special handling is required on EBCDIC platforms to get `qr/[i-j]/` to match only `"i"` and `"j"`, since there are 7 characters between the code points for `"i"` and `"j"`. This special handling had only been invoked when both ends of the range are literals. Now it is also invoked if any of the `\N{...}` forms for specifying a character by name or Unicode code point is used instead of a literal. See "*Character Ranges*" in *perlrecharclass*.

Internal Changes

- SVs of type `SVt_NV` are now bodyless when a build configure and platform allow it, specifically `sizeof(NV) <= sizeof(IV)`. The bodyless trick is the same one as for IVs since 5.9.2, but for NVs, unlike IVs, is not guaranteed on all platforms and build configurations.
- The `$DB::single`, `$DB::signal` and `$DB::trace` now have set and get magic that stores their values as IVs and those IVs are used when testing their values in `pp_dbstate`. This prevents perl from recursing infinity if an overloaded object is assigned to any of those

variables. [perl #122445]

- `Perl_tmps_grow` which is marked as public API but undocumented has been removed from public API. If you use `EXTEND_MORTAL` macro in your XS code to preextend the mortal stack, you are unaffected by this change.
- `cv_name`, which was introduced in 5.21.4, has been changed incompatibly. It now has a flags field that allows the caller to specify whether the name should be fully qualified. See "`cv_name`" in *perlapi*.
- Internally Perl no longer uses the `SVs_PADMY` flag. `SvPADMY()` now returns a true value for anything not marked `PADTMP`. `SVs_PADMY` is now defined as 0.
- The macros `SETsv` and `SETsvUN` have been removed. They were no longer used in the core since commit 6f1401dc2a, and have not been found present on CPAN.
- The `SvFAKE` bit (unused on HVs) got informally reserved by David Mitchell for future work on vtables.
- The `sv_catpv_n_flags` function accepts `SV_CATBYTES` and `SV_CATUTF8` flags, which specify whether the appended string is bytes or utf8, respectively.
- A new opcode class, `METHOP` has been introduced, which holds class/method related info needed at runtime to improve performance of class/object method calls.
`OP_METHOD` and `OP_METHOD_NAMED` are moved from being `UNOP/SVOP` to being `METHOP`.

Selected Bug Fixes

- Locking and unlocking values via `Hash::Util` or `Internals::SvREADONLY` no longer has any effect on values that are read-only to begin. Previously, unlocking such values could result in crashes, hangs or other erratic behaviour.
- The internal `looks_like_number` function (which `Scalar::Util` provides access to) began erroneously to return true for `"-e1"` in 5.21.4, affecting also `-'-e1'`. This has been fixed.
- The flip-flop operator (`..` in scalar context) would return the same scalar each time, unless the containing subroutine was called recursively. Now it always returns a new scalar. [perl #122829]
- Some unterminated `(?(...))` constructs in regular expressions would either crash or give erroneous error messages. `/(?(1)/` is one such example.
- `pack "w", $tied` no longer calls `FETCH` twice.
- List assignments like `($x, $z) = (1, $y)` now work correctly if `$x` and `$y` have been aliased by `foreach`.
- Some patterns including code blocks with syntax errors, such as `/ (? { (^ { }) }) /`, would hang or fail assertions on debugging builds. Now they produce errors.
- An assertion failure when parsing `sort` with debugging enabled has been fixed. [perl #122771]
- `*a = *b; @a = split //, $b[1]` could do a bad read and produce junk results.
- `In() = @array = split`, the `() =` at the beginning no longer confuses the optimizer, making it assume a limit of 1.
- Fatal warnings no longer prevent the output of syntax errors. [perl #122966]
- Fixed a NaN double to long double conversion error on VMS. For quiet NaNs (and only on Itanium, not Alpha) negative infinity instead of NaN was produced.

- Fixed the issue that caused `make distclean` to leave files behind that shouldn't. [perl #122820]
- AIX now sets the length in `getsockopt` correctly. [perl #120835], [rt #91183], [rt #85570].
- During the pattern optimization phase, we no longer recurse into GOSUB/GOSTART when not SCF_DO_SUBSTR. This prevents the optimizer to run "forever" and exhaust all memory. [perl #122283]
- `t/op/crypt.t` now performs SHA-256 algorithm if the default one is disabled. [perl #121591]
- Fixed an off-by-one error when setting the size of shared array. [perl #122950]
- Fixed a bug that could cause perl to execute an infinite loop during compilation. [perl #122995]

Acknowledgements

Perl 5.21.5 represents approximately 4 weeks of development since Perl 5.21.4 and contains approximately 40,000 lines of changes across 530 files from 33 authors.

Excluding auto-generated files, documentation and release tools, there were approximately 29,000 lines of changes to 390 .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 5.21.5:

Aaron Crane, Abigail, Alberto Simões, Andrew Fresh, Chris 'BinGOs' Williams, Craig A. Berry, Dagfinn Ilmari Mannsøker, Daniel Dragan, David Mitchell, Doug Bell, Ed J, Father Chrysostomos, George Greer, Graham Knop, James E Keenan, Jarkko Hietaniemi, Karen Etheridge, Karl Williamson, Leon Timmermans, Lukas Mai, Niko Tyni, Peter Martini, Petr Pá-sa™, Rafael Garcia-Suarez, Reini Urban, Ricardo Signes, Shlomi Fish, Steve Hay, syber, Tony Cook, Vincent Pit, Yves Orton, Årvar Arnfrø Bjarmason.

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.