NAME

perl5132delta - what is new for perl v5.13.2

DESCRIPTION

This document describes differences between the 5.13.2 release and the 5.13.1 release.

If you are upgrading from an earlier release such as 5.10, first read *perl5120delta*, which describes differences between 5.10 and 5.12.

Incompatible Changes

localised tied scalars are tied again.

The change in behaviour in 5.13.1 of localising tied scalar values has been reverted to the existing 5.12.0 and earlier behaviour (the change for arrays and hashes remains).

Naming fixes in Policy_sh.SH may invalidate Policy.sh

Several long-standing typos and naming confusions in Policy_sh.SH have been fixed, standardizing on the variable names used in config.sh.

This will change the behavior of Policy.sh if you happen to have been accidentally relying on the Policy.sh incorrect behavior. We'd appreciate feedback from anyone using Policy.sh to be sure nothing is broken by this change (c1bd23).

Stashes are now always defined

`defined %Foo::` now always returns true, even when no symbols have yet been defined in that package.

This is a side effect of removing a special case kludge in the tokeniser, added for 5.10.0, to hide side effects of changes to the internal storage of hashes that to drastically reduce their memory usage overhead.

Calling defined on a stash has been deprecated since 5.6.0, warned on lexicals since 5.6.0, and has warned for stashes (and other package variables) since 5.12.0. `defined %hash` has always exposed an implementation detail - emptying a hash by deleting all entries from it does not make `defined %hash` false, hence `defined %hash` is not valid code to determine whether an arbitrary hash is empty. Instead, use the behaviour that an empty `%hash` always returns false in a scalar context.

Core Enhancements

Non-destructive substitution

The substitution operator now supports a `/r` option that copies the input variable, carries out the substitution on the copy and returns the result. The original remains unmodified.

```
my $old = 'cat';
my $new = $old =~ s/cat/dog/r;
# $old is 'cat' and $new is 'dog'
```

This is particularly useful with `map`. See *perlop* for more examples (4f4d75, 000c65).

package block syntax

A package declaration can now contain a code block, in which case the declaration is in scope only inside that block. So `package Foo { ... }` is precisely equivalent to `{ package Foo; ... }`. It also works with a version number in the declaration, as in `package Foo 1.2 { ... }`. See *perlfunc* (434da3..36f77d, 702646).

CLONE_PARAMS structure added to ease correct thread creation

Modules that create threads should now create `CLONE_PARAMS` structures by calling the new function `Perl_clone_params_new()`, and free them with `Perl_clone_params_del()`. This will ensure compatibility with any future changes to the internals of the `CLONE_PARAMS` structure layout, and that it is correctly allocated and initialised.

perl -h no longer recommends -w

perl -h used to mark the -w option as recommended; since this option is far less useful than it used to be due to lexical 'use warnings' and since perl -h is primary a list and brief explanation of the command line switches, the recommendation has now been removed (60eaec).

Modules and Pragmata

Updated Modules

Locale-Codes 3.13

Locale::Country, Locale::Language and Locale::Currency were updated from 3.12 to 3.13 of the Locale-Codes distribution to include locale code changes (e1137b).

Thread-Semaphore 2.11

Added new methods ->down_nb() and ->down_force() at the suggestion of Rick Garlick.

Refactored methods to skip argument validation when no argument is supplied.

(04febe, f06daa)

CPAN.pm 1.94_57

* release 1.94_57

* bugfix: treat modules correctly that are deprecated in perl 5.12.

* bugfix: RT #57482 and #57788 revealed that configure_requires implicitly assumed build_requires instead of normal requires. (Reported by Andrew Whatson and Father Chrysostomos respectively)

* testfix: solaris should run the tests without expect because (some?) solaris have a broken expect

* testfix: run tests with cache_metadata off to prevent spill over effects from previous test runs

(742adb)

Hash::Util warning fix

Hash::Util now enables "no warnings 'uninitialized'" to suppress spurious warnings from undefined hash values (RT #74280).

B::Deparse now handles 'no VERSION'

The 'no 5.13.2' or similar form is now correctly handled by B::Deparse.

IO::Socket doc additions

getsockopt and setsockopt are now documented.

B::Concise updated for OPpDEREF

B::Concise marks rv2sv, rv2av and rv2hv ops with the new OPpDEREF flag as "DREFed".

File::Copy doc clarification

An extra stanza was added explaining behaviours when the copy destination already exists and is a directory.

Multiple POD spelling fixes.

Fixes were made to VMS::DCLsym, mro, Search::Dist, B::t::OptreeCheck and UNIVERSAL.

Changes to Existing Documentation

Replace wrong tr/// table in perlebcdic.pod

perlebcdic.pod contains a helpful table to use in tr/// to convert between EBCDIC and Latin1/ASCII. Unfortunately, the table was the inverse of the one it describes, though the code that used the table worked correctly for the specific example given.

The table has been changed to its inverse, and the sample code changed to correspond, as this is easier for the person trying to follow the instructions since deriving the old table is somewhat more complicated.

The table has also been changed to hex from octal, as that is more the norm these days, and the recipes in the pod altered to print out leading zeros to make all the values the same length, as the table that they can generate has them (5f26d5).

Document tricks for user-defined casing

perlunicode.pod now contains an explanation of how to override, mangle and otherwise tweak the way perl handles upper, lower and other case conversions on unicode data, and how to provide scoped changes to alter one's own code's behaviour without stomping on anybody else (71648f).

Document $# and $* as removed and clarify $#array usage

$# and $* were both disabled as of perl5 version 10; this release adds documentation to that effect, a description of the results of continuing to try and use them, and a note explaining that $# can also function as a sigil in the $#array form (7f315d2).

INSTALL explicitly states the requirement for C89

This was already true but it's now Officially Stated For The Record (51eec7).

No longer advertise Math::TrulyRandom

This module hasn't been updated since 1996 so we can't recommend it any more (83918a).

perlfaq synchronised to upstream

The FAQ has been updated to commit 37550b8f812e591bcd0dd869d61677dac5bda92c from the perlfaq repository at git@github.com:briandfoy/perlfaq.git

Performance Enhancements

Only allocate entries for @_ on demand - this not only saves memory per subroutine defined but should hopefully improve COW behaviour (77bac2).

Multiple small improvements to threads

The internal structures of threading now make fewer API calls and fewer allocations, resulting in noticeably smaller object code. Additionally, many thread context checks have been deferred so that they're only done when required (although this is only possible for non-debugging builds).

Size optimisations to SV and HV structures

xhv_fill has been eliminated from struct xpvhv, saving 1 IV per hash and on some systems will cause struct xpvhv to become cache aligned. To avoid this memory saving causing a slowdown elsewhere, boolean use of HvFILL now calls HvTOTALKEYS instead (which is equivalent) - so while the fill data when actually required is now calculated on demand, the cases when this needs to be done should be few and far between (f4431c .. fcd245).

The order of structure elements in SV bodies has changed. Effectively, the NV slot has swapped location with STASH and MAGIC. As all access to SV members is via macros, this should be completely transparent. This change allows the space saving for PVHVs documented above, and may reduce the memory allocation needed for PVIVs on some architectures.

Optimisation of regexp engine string comparison work

The foldEQ_utf8 API function for case-insensitive comparison of strings (which is used heavily by the regexp engine) was substantially refactored and optimised - and its documentation much improved as a free bonus gift (8b3587, e6226b).

Memory consumption improvements to Exporter

The @EXPORT_FAIL AV is no longer created unless required, hence neither is the typeglob backing it - this saves about 200 bytes per Exporter using package that doesn't use this functionality.

Installation and Configuration Improvements

Compilation improvements

Fix CCINCDIR and CCLIBDIR for mingw64 cross compiler to correctly be under $(CCHOME)\mingw\include and \lib rather than immediately below $(CCHOME).

This means the 'incpath', 'libpth', 'ldflags', 'lddlflags' and 'ldflags_nolargefiles' values in Config.pm and Config_heavy.pl are now set correctly (23ae7f).

Selected Bug Fixes

* Timely cleanup of SVs that are cloned into a new thread but then discovered to be orphaned (i.e. their owners are -not- cloned) (e42956)

* Don't accidentally clone lexicals in scope within active stack frames in the parent when creating a child thread (RT #73086) (05d04d).

* Avoid loading feature.pm when 'no 5.13.2;' or similar is encountered (faee19).

* Trap invalid use of SvIVX on SVt_REGEXP when assertions are on (e77da3)

* Don't stamp on $DB::single, $DB::trace and $DB::signal if they already have values when $^P is assigned to (RT #72422) (4c0f30).

* chop now correctly handles perl's extended UTF-8 (RT #73246) (65ab92)

* Defer signal handling when shared SV locks are held to avoid deadlocks (RT #74868) (65c742).

* glob() no longer crashes when %File::Glob:: is empty and CORE::GLOBAL::glob isn't present (4984aa).

* perlbug now always permits the sender address to be changed before sending - if you were having trouble sending bug reports before now, this should fix it, we hope (e6eb90).

* Overloading now works properly in conjunction with tied variables. What formerly happened was that most ops checked their arguments for overloading *before* checking for magic, so for example an overloaded object returned by a tied array access would usually be treated as not overloaded (RT #57012) (6f1401, ed3b9b, 6a5f8c .. 24328f).

* Independently, a bug was fixed that prevented $tied->() from always calling FETCH correctly (RT #8438) (7c7501)

Changed Internals

* The implementation of sv_dup_inc() has changed from a macro to a function.

- The `find_rundefsvoffset` function has been deprecated. It appeared that its design was insufficient to reliably get the lexical $_ at run-time.

  Use the new `find_rundefsv` function or the `UNDERBAR` macro instead. They directly return the right SV representing $_, whether it's lexical or dynamic (789bd8 .. 03d5bc).

- The following new functions or macros have been added to the public API: `SvNV_nomg`, `sv_2nv_flags`, `find_rundefsv`.

- The `UNDERBAR` macro now calls `find_rundefsv`. `dUNDERBAR` is now a noop but should still be used to ensure past and future compatibility.

- The ibcmp_* functions have been renamed and are now called foldEQ, foldEQ_locale and foldEQ_utf8 (e6226b).

Deprecations

The following items are now deprecated.

- Omitting a space between a regex pattern or pattern modifiers and the following word is deprecated. For example, `m/foo/sand $bar` will still be parsed as `m/foo/s and $bar` but will issue a warning.

Platform Specific Notes

Recent OpenBSDs now use perl's malloc

OpenBSD > 3.7 has a new malloc implementation which is mmap based and as such can release memory back to the OS; however for perl using this malloc causes a substantial slowdown so we now default to using perl's malloc instead (RT #75742) (9b58b5).

Acknowledgements

Perl 5.13.2 represents thirty days of development since Perl 5.13.1 (and two days of waiting around while the release manager remembered where he left his brain) and contains 3685 lines of changes across 194 files from 30 authors and committers.

Thank you to the following for contributing to this release:

Abigail, Andreas J. Koenig, Chas. Owens, Chris 'BinGOs' Williams, Craig A. Berry, David Caldwell, David Golden, David Mitchell, Father Chrysostomos, George Greer, H.Merijn Brand, Jerry D. Hedden, Karl Williamson, Maik Hentsche, Matt S Trout, Nicholas Clark, Rafael Garcia-Suarez, Ricardo Signes, Salvador Fandino, Salvador Ortiz Garcia, Shlomi Fish, Sinan Unur, Sisyphus, Slaven Rezic, Sullivan Beck, Tony Cook, Vincent Pit, Zefram, brian d foy, Ã†var ArnfjÃ¶rÃ° Bjarmason

Your humble release manager would like to specifically call out Karl Williamson for making the tests a better place to be, and Shlomi Fish for a passel of tiny incremental docfixes of the sort that don't get made often enough.

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 http://rt.perl.org/perlbug/ . 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 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.