

NAME

perl52111delta - what is new for perl v5.21.11

DESCRIPTION

This document describes differences between the 5.21.10 release and the 5.21.11 release.

If you are upgrading from an earlier release such as 5.21.9, first read *perl52110delta*, which describes differences between 5.21.9 and 5.21.10.

Deprecations

Making all warnings fatal is discouraged

The documentation for *fatal warnings* notes that `use warnings FATAL => 'all'` is discouraged and provides stronger language about the risks of fatal warnings in general.

Modules and Pragmata

Updated Modules and Pragmata

- *attributes* has been upgraded from version 0.26 to 0.27.
- *B* has been upgraded from version 1.57 to 1.58.
- *B::Deparse* has been upgraded from version 1.34 to 1.35.
`<<>>` is now correctly deparsed.
- *Config::Perl::V* has been upgraded from version 0.23 to 0.24.
- *CPAN* has been upgraded from version 2.10 to 2.11.
- *Cwd* has been upgraded from version 3.55 to 3.56.
- *ExtUtils::Miniperl* has been upgraded from version 1.04 to 1.05.
- *ExtUtils::ParseXS* has been upgraded from version 3.27 to 3.28.
- *ExtUtils::Typemaps* has been upgraded from version 3.25 to 3.28.
- *File::Spec* has been upgraded from version 3.55 to 3.56.
- *if* has been upgraded from version 0.0603 to 0.0604.
The documentation now notes that this module also works with `no`.
- *IO::Socket::IP* has been upgraded from version 0.36 to 0.37.
- *Module::CoreList* has been upgraded from version 5.20150320 to 5.20150420.
Updated to cover the latest releases of Perl. Also fixes a fencepost error in `is_core()` and copes with versions that do not match `x.yyyzzz`.
- *perl5db.pl* has been upgraded from 1.48 to 1.49.
The debugger would cause an assertion failure. [*perl #124127*]
- *PerlIO::mmap* has been upgraded from version 0.013 to 0.014.
- *utf8* has been upgraded from version 1.15 to 1.16.
Minor documentation update only.
- *warnings* has been upgraded from version 1.31 to 1.32.

Documentation

Changes to Existing Documentation

perlebcdic

- This document has been significantly updated in the light of recent improvements to EBCDIC support.

perlfunc

- Mention that `study()` is currently a no-op.

perlguts

- The OOK example has been updated to account for COW changes and a change in the storage of the offset.

perlhacktips

- Documentation has been added illustrating the perils of assuming the contents of static memory pointed to by the return values of Perl wrappers for C library functions doesn't change.

perlport

- Out-of-date VMS-specific information has been fixed/simplified.

perluniintro

- Advice for how to make sure your strings and regular expression patterns are interpreted as Unicode has been revised to account for the new Perl 5.22 EBCDIC handling.

perlvms

- Out-of-date and/or incorrect material has been removed.

Testing

- *t/porting/re_context.t* has been added to test that *utf8* and its dependencies only use the subset of the `$1..$n` capture vars that `Perl_save_re_context()` is hard-coded to localize, because that function has no efficient way of determining at runtime what vars to localize.

Platform Support

Platform-Specific Notes

Win32

miniperl.exe is now built with `-fno-strict-aliasing`, allowing 64-bit builds to complete on GCC 4.8. [*perl #123976*]

`test-prep` again depends on `test-prep-gcc` for GCC builds. [*perl #124221*]

Internal Changes

- Perl 5.21.2 introduced a new build option, `-DPERL_OP_PARENT`, which causes the last `op_sibling` pointer to refer back to the parent rather than being `NULL`, and where instead a new flag indicates the end of the chain. In this release, the new implementation has been revised; in particular:
 - On `PERL_OP_PARENT` builds, the `op_sibling` field has been renamed `op_sibparent` to reflect its new dual purpose. Since the intention is that this field should primarily be accessed via macros, this change should be transparent for code written to work under `PERL_OP_PARENT`.
 - The newly-introduced `op_lastsisib` flag bit has been renamed `op_moresib` and its logic inverted; i.e. it is initialised to zero in a new op, and is changed to 1 when an op gains a sibling.
 - The function `Perl_op_parent` is now only available on `PERL_OP_PARENT` builds. Using it on a plain build will be a compile-time error.

- Three new macros, `OpMORESIB_set`, `OpLASTSIB_set`, `OpMAYBESIB_set` have been added, which are intended to be a low-level portable way to set `op_sibling / op_sibparent` while also updating `op_moresib`. The first sets the sibling pointer to a new sibling, the second makes the op the last sibling, and the third conditionally does the first or second action. The `op_sibling_splice()` function is retained as a higher-level interface that can also maintain consistency in the parent at the same time (e.g. by updating `op_first` and `op_last` where appropriate).
- The macro `OpSIBLING_set`, added in Perl 5.21.2, has been removed. It didn't manipulate `op_moresib` and has been superseded by `OpMORESIB_set` et al.
- The `op_sibling_splice` function now accepts a null `parent` argument where the splicing doesn't affect the first or last ops in the sibling chain, and thus where the parent doesn't need to be updated accordingly.

Selected Bug Fixes

- `pack("D", $x)` and `pack("F", $x)` now zero the padding on x86 long double builds. GCC 4.8 and later, under some build options, would either overwrite the zero-initialized padding, or bypass the initialized buffer entirely. This caused `op/pack.t` to fail. [*perl #123971*]
- Extending an array cloned from a parent thread could result in "Modification of a read-only value attempted" errors when attempting to modify the new elements. [*perl #124127*]
- An assertion failure and subsequent crash with `*x=<y>` has been fixed. [*perl #123790*]
- An optimization for state variable initialization introduced in Perl 5.21.6 has been reverted because it was found to exacerbate some other existing buggy behaviour. [*perl #124160*]
- The extension of another optimization to cover more ops in Perl 5.21 has also been reverted to its Perl 5.20 state as a temporary fix for regression issues that it caused. [*perl #123790*]
- New bitwise ops added in Perl 5.21.9 accidentally caused `$^H |= 0x1c020000` to enable all features. This has now been fixed.
- A possible crashing/looping bug has been fixed. [*perl #124099*]
- UTF-8 variable names used in array indexes, unquoted UTF-8 HERE-document terminators and UTF-8 function names all now work correctly. [*perl #124113*]
- Breakage in `require(VERSION)` (with parentheses), introduced in Perl 5.21.6, has been fixed. [*perl #124135*]
- Wide char warnings, wrongly raised in UTF-8 locales since Perl 5.21.8, are now suppressed. [*perl #123527*]
- A subtle bug introduced in Perl 5.21.4 involving UTF-8 in regular expressions and sometimes causing a crash has been fixed. A new test script has been added to test this fix; see under *Testing*. [*perl #124109*]
- A change introduced in Perl 5.21.10 that broke the autovivification CPAN module has been reverted.
- A bug whereby `'FOO' -> f()` lost the read-only status of `FOO`, introduced in Perl 5.21.7, has been fixed. [*perl #123619*]
- A regression in Perl 5.21 in the matching of `/[A-Z]/ai` has been fixed.

Acknowledgements

Perl 5.21.11 represents approximately 4 weeks of development since Perl 5.21.10 and contains approximately 5,800 lines of changes across 240 files from 24 authors.

Excluding auto-generated files, documentation and release tools, there were approximately 1,700 lines of changes to 160 .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 5.21.11:

Alex Vandiver, Andreas Knig, Craig A. Berry, Dagfinn Ilmari Mannsker, Daniel Dragan, David Golden, David Mitchell, Father Chrysostomos, H.Merijn Brand, Herbert Breunung, Hugo van der Sanden, James E Keenan, James McCoy, Karl Williamson, Matthew Horsfall, Max Maischein, Nicholas Clark, Rafael Garcia-Suarez, Randy Stauner, Ricardo Signes, Shirakata Kentaro, Steffen Mller, Steve Hay, Tony Cook.

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.