NEWS for R Under development (unstable) (2026-11-16 r90911)



Changes in R-devel

Significant User-Visible Changes:

- `read.dcf()` and `write.dcf()` now treat DCF files (such as package '`DESCRIPTION`' and repository '`PACKAGES`' files) as UTF-8, as required by the Debian control file format. `read.dcf()` returns values in UTF-8 and `write.dcf()` always writes UTF-8 regardless of the session locale; bytes which are not valid UTF-8 are preserved by escaping them as '`<xx>`' rather than being re-encoded via a '`Encoding`' field. As a consequence, the `useBytes` argument of `write.dcf()` is ignored, and non-UTF-8 '`DESCRIPTION`' files (e.g. those declaring '`Encoding: latin1`') are no longer re-encoded on reading.
- Fixing `rbinom()`, see PR#19049 below, to reproduce old results needs `RNGkind()` to get a new `binom.kind` argument and, e.g. `RNGversion("4.6")` keeps previous behaviour.
- The Sweave syntax definition objects `SweaveSyntaxNoweb` and `SweaveSyntaxLatex` now each have their own detailed help page.
- The translations have been updated by the R Translation Team, with the patch generated by Gergely Daroczi. This includes a major contribution of over 4000 Arabic translations from Abdulrahman Alswaji.

New Features:

- `boxplot()`: when `notch=TRUE`, new option `warnN=FALSE` suppresses warnings about "notches outside hinges".
- New function `cumvar()` to calculate cumulative variances.
- The `print()` method for `rle()` now passes `...` to `str()`.
- `rmultinom()` now uses Kahan's arithmetic instead of `LDOUBLE` to avoid platform dependency, thanks to Mauricio Vargas' proposal in PR#18693.
- *Object not found* errors now signal error conditions of class `objectNotFoundError`. *Function not found* errors signal conditions of the subclass `functionNotFoundError`.
- Condition classes used internally for signaling some errors are now documented in `?conditions`.

- The `as.data.frame()` methods for atomic `vector` classes get an optional switch `validRN`, with default true to fix PR#19059 and false for some back compatibility, still allowing data frames in construction to contain `NA`s in row names.
- `tools::Rd2txt()` now replaces '`---`' and '`--`' by Unicode dashes when the output encoding is UTF-8.
- `fisher.test()` gains a new option `pval.only = TRUE` for the 2x2 exact case, with substantial speed gains in the case of large counts.
- `binom.test()` and `poisson.test()` can now at least optionally compute central two-sided p-values which match the confidence intervals. Same for `fisher.test()` in the 2×2 case and `mantelhaen.test()` in the $2 \times 2 \times K$ case if an exact conditional test is performed.
- New low-level function `.OBJSXP()` enables deparsing of non-S4 `"object"` type objects.
- `use()` gets an argument `pos` with a help example showing how to attach a "chain" of packages to `search()`.
- New higher-order functions `Compose()`, `Funcall()` and `Partial()`.
- New functions `find_calls()`, `find_calls_in_package_code()`, `gsub_with_transformed_matches()` and `system_with_capture()` in package **tools**.
- `data()` will accept text files with further extension '`.zst`' or '`.zstd`', and read them on builds with `zstd` support.
  `free1way(, MPL_Jeffreys = TRUE)` implements maximum penalised likelihood estimation with Jeffreys prior for all model parameters simultaneously, along with Wald-type inference.
- `rfree1way` also accepts `offset` terms equal to the number of simulated observations, allowing to sample from more complex models.
- `tools::add_datalist()` now returns a logical (invisibly, instead of `NULL`), indicating if it did add a '`data/datalist`' file (addressing parts of PR#18015).
- `gzcon` `rawConnection`s now can flush and are writable thanks to Ivan Krylov's analysis and patch in PR#19172.
- `chol2inv()` gains optional switch `diag.only` for shortcutting `diag(chol2inv(.))`, wished for with a patch and reproducible analysis in PR#19177 by Adam Marstrand.
- Dynamic HTML help pages are postprocessed to link function arguments in the Usage section to their documentation in the Arguments section.
- The X11 on-screen devices now supports history navigation through (customizable) key bindings.
- `model.matrix.default()` now catches and names an unsuitable factor before calling `contrasts<-` (PR#19148), and no longer warns about unused elements of `contrasts.arg` (PR#18313, obviating an all too common workaround using `suppressWarnings`).
- `as.vector` gains methods for classes `Date`, `POSIXct`, and `difftime` that call methods for `as.list` or `as.character` when `mode="list"` or `"character"`. The method for class `POSIXlt` is changed to do the same and for all other modes work via class `POSIXct`, a corollary being that `as.vector(., mode="double")` now works as `as.double(.)` rather than signalling an error.

Installation on a Unix-Alike:

- Installing `R` (creating '`libR.pc`') no longer fails with a `sed` error when the new '`--package-metadata`' linker flag is involved; an '`s`'-command delimiter other than '`:`' was needed (PR#19060, thanks to Jeroen Ooms and Dirk Eddelbuettel).

- `libzstd` is required. For R 4.7.0 only, this can be circumvented by configuring with '`--without-libzstd`'.

Package Installation:

- .

C-Level Facilities:

- .

Utilities:

- `tools::texi2pdf()` now reports BibTeX warnings; in particular, missing references are now noted by `R CMD check` when re-building PDF vignettes.
- `R CMD build` now excludes an obsolete '`data/datalist`' file when the package uses '`LazyData`', and reports when it added one.
- The long-standing environment variable `PKG_BUILD_DIR` to control where `R CMD INSTALL` unpacks source tarballs is now documented in the R Internals manual. This is supplemented by `PKG_BUILD_DIR_BARE` to use `PKG_BUILD_DIR` as the unpacking directory without adding a random path component.
- `R CMD check` now '`NOTE`'s packages listed more than once in a dependency field other than to give a version range, e.g., '`Imports: utils, stats, utils`'.

Deprecated and Defunct:

- `structure()` now deprecates the old "special" substitute `attributes`' names `".Dim", ".Dimnames", ".Names", ".Tsp", ".Label"` in favour of the respective `"dim", "dimnames", "names", "tsp", "levels"` names.

Bug Fixes:

- For `x <- <large-POSIXlt>`, for some boundary cases with `x$sec - floor(x$sec)` close to 1, `c(x)` could become one second ahead of `x` in R 4.6.0; subassignments such as `x[i] <- val` have improved, thanks to suggestions by Suharto Anggono in PR#18989.
- In an R installation without NLS, e.g., from '`--disable-nls`' configuration, the C `ngettext()` replacement now also keeps plural for "0 things", thanks to Michael Chirico's PR#19065.
- `as.data.frame.vector(<row.names_w/_NA>)` now signals the same error as `data.frame()`, fixing Hervé Pagès' PR#19059. From Suharto Anggono's comment #c4, it gives the same error also when `row.names = RN` explicitly, where `RN` has `NA`s.
- `.leap.seconds` again has no `"tzone"` attribute, as in earlier R versions, fixing PR#19072 thanks to Suharto Anggono.
- `as.POSIXct(x)` for `POSIXlt` object `x` with `.Machine$integer.max - x$year < 1900` no longer suffers from C level integer overflow, thanks to Suharto Anggono's patch for PR#19038. Currently, this does not apply when the `"internal"` code for date-time conversion and printing is in use, as e.g., on Windows and macOS.
- `wilcox.test()` gains new optional argument `digits.zap`, *and* the default for argument `digits.rank` is changed from `Inf` to `7`, addressing bugs PR#18035 and PR#19071, in order to deal with both ties of almost equal numbers and almost zeros of differences.
- Exclude patterns from '`.Rbuildignore`' are now processed *before* preparing the package directory for the tarball, avoiding redundant file copies from the source directory (PR#17549). In particular, when a source package '`foo/`' folder contains broken symlinks matching an exclude pattern, `R CMD build foo` now succeeds, omitting these, thus fixing PR#18186. (Thanks to Ivan Krylov and others, notably Jenny Bryan and R Dev Day @ useR! 2025 contributors.)

- `strat_ratio != 1` in `rfree1way` did not work as advertised.
- Zeros in the `prob` argument to `power.free1way.test` triggered an error and are now handled correctly.
- Fix for PR#18874; `"null"` widths/heights in **grid** layouts are zero if widths/heights to allocate are zero. (Thanks to Maciej Nasinski and Trevor Davis.)
- Fix for PR#19055; additional documentation for `embedFonts()`. (Thanks to Chris Danek and Ivan Krylov.)
- `merge.data.frame` now returns the correct column names when at least one of the merged data frames has zero rows and at least one of the merged data frames has only one column, fixing PR#18930. Thanks to David Molnar and Gabriel Becker.
- `trace()` now works for functions whose S3 class attribute has more than one string, e.g., **S7** generics and methods. Thanks to Hadley Wickham for the patch.
- `as(object, to) <- value` no longer treats `to` as a class union when it was registered via `setOldClass()`.
- In **methods**, `setOldClass(..., S4Class = )` no longer warns about undefined subclasses when the supplied S4 class inherits from a class union.
- Fix for PR#19094; fill/stroke path based on text with `col=NA`. Thanks to Trevor Davis.
- Fix for PR#18842; better documentation and warnings when specify inappropriate `col` to `filled.contour()`. Thanks to Darakhshan Nehal.
- `cor(*, method="kendall")` and `cov()` now use the fast N*log(N) algorithm suggested in PR#18934 and offered by David Simcha on R-devel in 2010. With thanks to the R Dev Day @ Nantes 2026 contributors, notably Charles Bordet and Alexis Derumigny.
- In `nls`, follow suggestion by Ben Bolker to reorder named vectors for `lower` and `upper` arguments to match names of the starting values (PR#18415). This has been extended so that you can now specify *only* the parameters that you wish to constrain, and also so that it works when parameters are in the form of named lists.
- `summary(<aov>)` for an almost perfect fit now warns analogously to `summary(<lm>)`, using the same check. Fixes fixes PR#18341 thanks to 'R Dev Day @ useR! 2026' participants Deepansh Khurana, Stefan Linner and Oscar de Leon.
- Package **methods** provides a method for implicit generic function `rcond` with signature `c(x="ANY", norm="missing")` so that `rcond(x)` does not break after `setGeneric("rcond")`.
- `rbinom()`'s BTPE algorithm is fixed thanks to Kate Hudson's findings and report in PR#19049. To ensure reproducibility, `RNGkind()` gets new argument `binom.kind`.
- Fix segfault in `postscript()` and `pdf()` if set `papersize` option to very long string (PR#19108). Thanks to Kevin Ushey.
- Fix segfault in `grid::gpar()` if set `fontfamily` longer than 200 characters (PR#19107). Thanks to Kevin Ushey.
- Fix cases where `M` and `X` matrices could be spanning the same subspace in `mauchly.test()`, `sphericity()` `anova.mlm()`, `anova.mlmlist()`. (PR#16166, originally reported in 2015 by Florent Aubry, essentially analyzed Feb. 2020 just before the corona crisis, triaged 2026 by Michael Chirico)
- Changed `model.frame.glm()` to use same logic as `model.frame.lm()` (PR#19036, Isaac Gravestock).
- `drop1.glm()` misbehaved for models fitted with `y = FALSE` (PR#19128, Amal Tawfik).
- `confint(<free1way>, parm)` returned confidence intervals for all parameters when `parm` requested a subset.

- `summary(<free1way>)` renames p-values to `Pr(..)` and thus `printCoefmat` recognises them.
- Fix for PR#19035 (piping `postscript()` output) thanks to Jeroen Ooms and Ivan Krylov.
- `lm(.., weights = *)` gets a new optional argument `wtol`, defaulting to 0 currently for back compatibility. Setting `wtol = 1e-30` fixes PR#17475 thanks to the 'R Dev Day' 2025 participants Kelly Bodwin et al.
- `glm(*, wtol = <positive>)` treats very small weights as zeros via `glm.fit()` and `glm.control()`, also with new optional `tol` argument, more similar to `lm()`, but not equivalently, e.g., for computing `AIC()`.
- Atypical `lm()` calls now work better, e.g., `fm <- lm(rock); update(fm, . ~ . - perm)` and similar cases, fixing PR#17463 and PR#17476 and part of PR#1861, thanks to Ben Tyner, Elin Waring, Kevin Tappe and other Bugzilla contributors.
- `pkg2HTML(dir="pkg")` and other **tools** consuming the `Rd_db()` of a source package, e.g., the `undoc()` QC function (PR#18960, thanks to Michael Chirico), no longer fail from processing '`DESCRIPTION`'-based Rd macros such as `\packageTitle` when the current directory is not the source package's root directory.
- `glm()` with zero weights now provides correct AIC and `logLik()`, fixing PR#16008, thanks to Bill Dunlap and Mauricio Vargas.
- `glm(rock)` and `glm(data = rock)` work like `lm(..)` now.
- `confint(<free1way>, what = "OVL")` now includes one for shift effect confidence intervals including zero.
- `x[logi]`, with `logi` a logical index, now warns if `length(x)` is not a multiple of `length(logi)`, fixing PR#19155 thanks to several participants in two R Dev Days and the R Sprint in 2026.
- The `as.list()` method for class `"function"` now checks its argument and behaves better for a fake "function", fixing PR#19173 thanks to Mark Padgham's report and patch proposal.
- The `anova(.., test = tt)` methods for `"glm"` and `"glmlist"` now do partially match a string `tt`, as documented, thanks to Amal Tawfik's report and patch in PR#19179.
- Added protection against out-of-range values for graphical parameters `lty`, `lend`, and `ljoin` in **graphics** function calls (PR#19170). Thanks to Adam Berkeš.
- `cairo_pdf()` no longer silently ignores `grid.newpage()` calls after drawing a `glyphGrob` containing two or more glyphs inside a masked `viewport`. Thanks to Trevor Davis' patch proposal in PR#19075.
- `trace` with non-`NULL exit` replaced rematched S4 method definitions with non-rematched ones that evaluated `exit` twice, first on `.local` exit and again on method exit.
- When tracing rematched S4 methods, `trace` could insert a call `trace(.local, tracer=, exit=, ...)` with improper `tracer` and `exit` components. Thanks to Hao Wang for the report (PR#18707).
- Fixes for `grid.pack(row.after=)` (and `row.before` and `col.after` and `col.before`) and `grid.pack(dynamic=TRUE)`. Thanks to Athenkosi Nkonyeni for reports.
- `parse_Rd()` should no longer leak memory when errors are raised, thanks to Kevin Ushey's report and patch in PR#19184.
- `delete.response` inadvertently converted indices (attributes `response`, `offset`, `specials`) from integer to double. It also replaced `NULL` components of `specials` with `logical(0L)`.
- `delete.response` did not modify optional attribute `dataClasses`.

- When interaction terms or offsets were present, `drop.terms` and `[.terms` could return a `terms` object with invalid `predvars` and `dataClasses` attributes. One consequence was that `x[-j]` and `x[seq_len(n)[-j]]` could differ for objects `x` with `n` terms and indices `j` in `seq_len(n)`.
- The S4 methods for `initialize` with signature `.Object="oldClass"` or `.Object=".environment"` evaluated arguments of type `"symbol"` or `"call"` when dispatching next methods, causing misbehaviour of `new(Class, ...)` with symbols or calls in `....` Thanks to Alexis Sardá-Espinosa for the report (PR#16629).
- The S4 methods for `initialize` with signature `.Object="array"`, `.Object="matrix"`, or `.Object="ts"` did not give higher precedence to named components of `...`, violating the convention of the default method. Thanks to Dan Murphy for the report (PR#14323).
- Fix for long-standing bug in Cairo raster output that sometimes produced gaps between adjacent filled rectangles (PR#16721). Thanks to Dave Dixon for the original report and to Ella Kaye, George Stagg, Kevin Ushey, and Ivan Krylov for contributions to the fix.
- `str()`'s output puts a space (`" "`) after `"<some>:"` and `"attr(*, \"<....>\")="` consistently now, addressing PR#18719 by Dave Slager and Novica Nakov. This was achieved by no longer *pre*pending spaces, such that e.g., `str(pi)`'s output no longer starts with a space.
  This should have little effect on (`R CMD check`)s, as `R CMD Rdiff` does *not* show such differences.
- `E <- evalSource(..., cache = FALSE)` did not create S4 method tables in `E`, so `insertSource(E, ...)` did not trace S4 methods. Thanks to Takeshi Arabiki for the report (PR#15242).
- `aperm(x, perm)` and `t(x)` did not support arrays `x` of type `"expression"`.
- `validObject(x)` coerced `x` to simple superclasses unnecessarily. Now validity methods for simple superclasses are called with `x` as the argument. Thanks to Hervé Pagès for the report (PR#17944).
- `canCoerce` did not consider an uncached implicit coercion from the object class to a subclass. Thanks to Hervé Pagès for the report (PR#16423).
- `dev.set(1)` now returns the number and name of the device it opens, as documented, rather than `c("null device" = 1)`, thanks to Trevor Davis' report and patch in PR#19281.
- `str(x)` no longer prints to full internal precision when `x` has a class, is of type `double`, but `is.numeric(x)` is false, as e.g., for `"difftime"` objects `x`. Rather, `str.default()` obeys its default or explicit `digits.d` argument, fixing PR#19294.
- Fix for `contourLines()` and `contour()` with negative z values (PR#19279). Thanks to Kevin Ushey.
- `as.environment(x)` and `pos.to.env(x)` returned a list rather than an environment for numeric `x` of length greater than one. Now they signal an error. Thanks to Adam Berkeš for the report (PR#19306).
- `as(object, name)` signalled an error when `class(object)[1]` was not formally defined and `class(object)[-1]` contained `name`, never searching for a method for `coerce` with signature `c(from="ANY", to=name)`. Now it searches and signals an error only on failure. Thanks to Michael Chirico for the report (PR#19023).
- The internal HTTP server (used by the dynamic help system) removes dot segments (`/../` and `/./`) also when they were percent-encoded and on Windows converts `\` to the `/` path separator. Paths that include `%00` are rejected.
- The internal HTTP server accepts request and header lines of up to 16k bytes (previously 1k) so that requests with many cookies no longer fail. This matters since

browsers also send cookies set by any other web application running on '`localhost`', as cookies are not specific to a port.

- The internal HTTP server no longer decodes '+' as a space in the request path (only in the query string) and no longer removes dot segments from the query string.
- The internal HTTP server now handles line endings split across network packets, responds with an error instead of waiting when a request or header line is too long, no longer corrupts request headers over 1k bytes in total and drops requests that were closed by the client before they were complete.
- The internal HTTP server now sends 1xx, 204 and 304 responses without a body (as required by RFC 9110), skips empty response headers and responds with an error when a handler returns an invalid status code.
- The internal HTTP server now rejects requests with ambiguous framing (as required by RFC 9112): `Transfer-Encoding` (not supported), invalid or conflicting `Content-Length`, whitespace before the colon in a header field and more than one `Host` header. Previously such requests could be misread.



Changes in R 4.6.2 patched

New Features:

- `capabilities()` now reports if `zstd` compression is supported.

Installation:

- `configure` now reports if `zstd` support was found.
- `configure` issues a warning if no way is found to determine C dependencies, as without this updating a build is unreliable, for example if '`Rversion.h`' is changed (and it holds the SVN revision). (Seen with macOS SDK 27.2.)
- The way `configure` tests for C and Objective-C `make` dependencies is now less fragile and works on macOS 27.2.

Bug Fixes:

- `unzip(., unzip = "internal")` no longer leaks the zip handle on error, thanks to Kevin Ushey, see PR#19233
- The internal method of `untar()` now checks the whole path of every member, so that a final component of `".."` or `"~"` can no longer place a link or file outside the extraction directory; rejects hard- and symbolic-link targets restored as file copies, which resolve outside the extraction directory or pass through a symbolic link; applies the `umask` to extracted files; also no longer restores '`setuid`' or '`setgid`' permission bits. Similarly, the internal method of `unzip()` no longer extracts through symbolic links below `exdir`, both thanks to Kevin Ushey's report and patch in PR#19229.
- `smooth(*, "S")` now reports a change when the last plateau it splits is unchanged on its right side, so "3RS3R" and "3RSS" are correct now in such cases. Fixes PR#19280 reported with patch by Kevin Ushey.
- `runmed(<has_NA_not_at_pos_1>, na.action = "na.omit")` no longer returns a "random" smoothed value before the `NA`, fixing PR#19300, from the report and patch by Adam Berkeš.



Changes in R 4.6.2

Bug Fixes:

- In plain-text help (`Rd2txt`), paragraphs are again recognized even when started with a non-indented Rd comment, fixing a regression in R 4.4.0.
- `rhyper(N, m, n, k)` now switches to the "large" scenario when `m + n` is larger than `.Machine$integer.max`, preventing integer overflow, thanks to Jonathan Blood's report on R-devel.
- Fix long standing (Fortran compiler optimization related) bug in `nlminb` with constraints, thanks to Kasper Kristensen's PR#19086 report and patch.
- Fix for PR#19100: `seq.int` was not handling non-vector `along.width` arguments properly. Thanks to Mitchell O'Hara-Wild.
- Wrappers now duplicate even a wrapped object that might be shared before returning a read-only pointer. This prevents cached read-only pointers from becoming invalid. Reported by Ivan Krylov.
- `dummy.coef()` becomes more robust to non-syntactic variable names, fixing PR#18468. Thanks to Bill Dunlap and 'R Dev Day' participants Vincent Arel-Bundock, Cynthia Huang, Tina Rozsos and Marina Marinelli.
- Now `PROTECT()` in `SubassignTypeFix`, fixing PR#19110 thanks to Kevin Ushey.
- `writeChar()` in a multibyte locale counts characters and bytes consistently and no longer overflows, thanks to Kevin Ushey's report and patch in PR#19112.
- `dim(a) <- vec` now checks for too large `prod(vec)`, thanks to Kevin Ushey's report and patch in PR#19109.
- Tweaks (including to the parsers) for the C23 behaviour (preserving `const`) of `strchr`, `strrchr` and `strstr`. (By Brian Ripley and Duncan Murdoch.)
- `make check-devel` now works without recommended packages.
- `<matrix>[[i, j]]` with negative `i` occasionally gives no error but accesses `previous content`; thanks to Kevin Ushey's report and very simple fix in PR#19116.
- Quartz device now correctly draws long paths (work-around for Quartz built-in stroker breaking long paths). Thanks to R Dev Day @ useR! 2026 contributors, notably Tomek Gieorgijewski, report in PR#18680.
- `strptime(ch, "...%OS<n>...")` now disregards contiguous digits in "<n>", for symmetry with `strftime()`, fulfilling Frederick Eaton's wish in PR#17209, thanks to a patch by Suharto Anggono.
- `strptime()` now also returns `NA` for out of range `"%OS"` and a correct error message in some wrong multibyte cases, fixing PR#19117 comment #1 and PR#19122 by Michael Chirico; additionally, some overflow to `Inf` times no longer result in `NA`.
- `strptime("0", "%w")` in a `"C"` locale now works in valid cases, returning `NA` for invalid week numbers (`"%U"` or `"%W"`), fixing PR#19124 thanks to Michael Chirico's report and notably Suharto Anggono's analysis, also correctly fixing PR#15195.
- `ksmooth(x,y, x.points=NULL)` now returns an empty (x,y) pair instead of crashing R, fixing PR#19153.
  Additionally, also by Adam Berkeš, `x <- rep(999,0); ksmooth(x, x, x.points = 1)` no longer writes out of bounds, fixing PR#19171. Ditto for `ksmooth(x, x, n.points = 0)`, PR#19219.
- `besselI(1, 1e11)` and `besselK(..)` now signal a warning and return `NaN` instead of typically segfaulting, fixing PR#19175 thanks to Ben Bolker.
  Further, `besselK(1, -1e11)` and similar warn about too large `|nu|`, returning `NaN` rather than segfaulting, fixing PR#19196 thanks to Adam Berkeš.
- `text.default(.., col = numeric(0))` and similar no longer segfault in "instrumented" versions of R, fixing PR#19178 thanks to Jiang Yuancheng's `fusion-fuzz` and Ivan Krylov.

- `R CMD rtags` (on Unix-alikes) no longer fails from a bad escape if `sh` is `bash` or `busybox`.
- `capabilities("X11")` no longer incorrectly returns `TRUE` on a Unix-alike where X11 is not actually available when built by a C compiler without C23 `enum` base types, thanks to Trevor Davis' report and patch in PR#19180.
- `predict(smooth.spline(..), deriv = -99)` and similar invalid `deriv` values now signal errors instead of segfaulting, thanks to Adam Berkeš' report and proposal in PR#19190.
- `format(x, width = 2^31 -1)` for an non-ascii character `x` such as `x = "\u00e9"` now signals an error as the required width is too large, rather than segfaulting, thanks to Adam Berkeš' report in PR#19188.
- `influence.measures()` inadvertently wasn't updated along with `dffits()`, `dfbetas()`, `covratio()`. PR#19167 by Amal Tawfik.
- If `wilcox.test(..., exact=TRUE)`'s p-value is close to the 0 or 1 boundaries, the calculation no longer suffers severely from cancellation, thanks to Tasuku Kobayashi's reports and Ahmed Sattar Jabbar's suggestion in PR#19144.
- Long stringed error messages are now protected from premature garbage collection, fixing PR#19210, reported and analyzed by Yuancheng Jiang.
- `strsplit(str, perl=TRUE, bytes=TRUE)` no longer accesses invalid memory in case `str` is an invalid UTF-8 string, fixing PR#19212, thanks to Kevin Ushey.
- `isoreg(numeric(0))` no longer returns unallocated component `yc`, fixing PR#19224 thanks to Kevin Ushey.
- `KalmanLike(1:5, list(1,2))` and similar no longer segfault, fixing PR#19217 thanks to Kevin Ushey and Adam Berkeš.

Bug Fixes (Windows):

- `path.expand("~")` and similar now should work correctly on Windows when the user name is not in ASCII and neither `%HOME%` nor `%USER%` environment variables are set, fixing PR#18949 thanks to the report and patch from Ivan Krylov and Mauricio Vargas's R Dev Day work.
- `file.copy(*, date=TRUE)` no longer opens files for exclusive access when copying the file time. Consequently, the time *is* copied now and concurrent attempts to open the source file should work reliably, thanks to Kevin Ushey's report and patch in PR#19187.



Changes in R 4.6.1

New Features:

- The LaTeX class '`jss`' bundled in '`share/texmf/tex/latex/jss.cls`' has been updated to work with '`hyperref`' version 7.01q (2026/04/24).
- Plain-text (and legacy HTML) conversion for a simple Rd `\eqn{}` now handles `\geq`, `\leq`, `\neq` and `\ne` in addition to `\ge` and `\le` (PR#19070).

Bug Fixes:

- `abbreviate(<non-ASCII>)` now works better, using `iswlower()` and `*upper()` to treat "wide" characters, thanks to Kevin's PR#19058.
- `cut(1:11, {})` now signals an error for the invalid `breaks`, fixing PR#19057, thanks to Kevin Ushey.

- `all(<symbol>)` signals an error now, as some similar coercions; previously, the coercion of the symbol resulted in `NULL` internally, raised a warning, and could access illegal memory; thanks to Kevin Ushey's PR#19054.
- `stl()` now checks its 3 x 3 tuning parameters on the R level; `stl(*, s.window = 0)` now "works" instead dumping core, `stl(*, s.window = 1)` "works" correctly, too. Also, the `summary()` method mentions the robustness `weights` only if any robustness iterations happened.
  Further, `stl()` is now based on C code, translated from previous Fortran, triggered by a flang 22 compiler optimization bug (`-O2`).
- The border case `zapsmall(x, digits=Inf)` where `x` contains an `Inf` entry no longer returns all `NaN` but rather `x`. The improved help page now has more non-default examples.
- `wilcox.test(x, exact=FALSE, correct = k)`'s p-value computation got a missing `dnorm(z)` factor, and so did the 2-sample case, in addition to more fixes.
- The data part of `getGenerics()` listed package names for generic functions defined in more than one package. Now it lists generic function names, as intended. Thanks to Michael Chirico for report and patch proposal (PR#19080).
- Overly long `dyn.load()` error messages (C level `dlerror()`), should no longer corrupt its state, thanks to Ivan Krylov's PR#19029 report and patch.
- `as.matrix(<POSIXlt>)` again produces a *numeric* matrix, and the `c()` and sub-assignment methods for `POSIXlt` objects work correctly more often, e.g., without a `"tzone"` attribute, back porting a few of the PR#18989 fixes from R-devel, both thanks to Suharto Anggono.
- In **methods**, `as()` now correctly coerces S4 objects to S4 superclasses extending old-style classes, instead of returning only the S3 part.



Changes in R 4.6.0

Significant User-Visible Changes:

- The R citation now provides a DOI (10.32614/R.manuals): please update your bibliographies accordingly.
- The default C++ standard has been changed to C++20 where available (which it is on all known platforms from 2021 on): if not C++17 is used if available otherwise C++ is not supported (as before).
  (GCC 16 has also switched to C++20 as its default.)
  Packages can request C++17 if essential.
- The Sweave driver `Rtangle` no longer puts empty lines at the end of tangled scripts.

New Features:

- The default `AIC` and `BIC` methods now check whether the **methods** namespace is loaded rather than **stats4**. They still use `stats4::logLik` (and `stats4::nobs`), but packages no longer need to load **stats4** explicitly.
- `confint.default()` now also works on S4 objects, as long as suitable `coef()` and `vcov()` methods exist.
- `binomial(identity)` and `quasibinomial(identity)` now work without having to quote the argument.
- `str()` now shows more via `format()` for external pointers.
- `terms(<formula>, specials = *)` now treats non-syntactic specials more gracefully, thanks to Mikael Jagan's PR#18568.

- `help.start()` gets a `lib.loc` option and its default for `make.packages.html()` is enhanced to allow, e.g., fast access to `.Library` when that contains only base and recommended packages.
- In '`*.Rd`' help files, the `\linkS4class` macro now also works for links to other packages via `\linkS4class[<pkg>]{<myClass>}`, much nicer than the "workaround" `\link[<pkg>:<myClass>-class]{<myClass>}` necessary in earlier versions of R. Packages with the new syntax need to formally depend on '`R >= 4.6.0`'.
- `tools::analyze_license()` now also computes SPDX license identifiers, thanks to Thierry Onkelinx and LLuís Revilla.
- `read.dcf()` now recognizes lines starting with '`#`' as comment lines. By Dirk Eddelbuettel, Laurent Gatto and Hugo Gruson.
- `tcltk::tclVersion()` now only optionally includes the 'patch level', as Tcl full versions can be of the form `8.7a5` or `9.0b1` without three clear components.
- `untar(tar = "internal")` supports `extras = "-P"` to use unchanged the recorded file paths (as many external `tar` programs do).
- `extSoftVersion()[["zstd"]]` now reports the version of the `libzstd` compression library if available, thanks to Trevor Davis' patch proposal in PR#18914.
- New logical `option() quiet`, defaulting to false, set to `TRUE` by R's command-line option '`--quiet`' (and similar), now can be switched during an R session. Prompted by Dirk Eddelbuettel's proposal in PR#18913.
- Credentials for basic HTTP authentication with download method `"libcurl"` can now be provided in a '`netrc`' file. The path to the file can be specified in `options("netrc")`.
- `tools::deparseLatex()` gains a `math` argument, now used by the default `bibstyle` to format `bibentry()` fields with '`$...$`' syntax via Rd `\eqn` markup.
- `gzcon()` can now decompress a concatenation of compressed streams PR#18887.
- Bibliographic citations and references in Rd files can now be auto-generated from bibentries in bibliographic databases in R or BibTeX formats. See `RShowDoc("R-exts#Bibliographic-citations-and-references")` for more information.
- Arguments to `loadNamespace()` which will be ignored because the namespace is already loaded are now noted.
- `chkDots()`' optional argument `allowed` has been implemented thanks to Trevor Davis' proposal in PR#18936.
- New function `messageCondition()` to create structured `"message"` objects analogous to `warningCondition()` and `errorCondition()` (PR#18940, by Trevor Davis).
- `wilcox.test()` can now perform exact (conditional) inference in case of ties. Based on contributions by Torsten Hothorn.
- `wilcox.test()` can now optionally compute improved asymptotic p-values by including up to 3 correction terms of the Edgeworth series for the normal approximation.
- `barplot(.., orderH = *)` for stacked bar plots allows to sort the stacks each according to size.
- New argument `panel.first` for `hist()` (via `plot.histogram()`), `barplot()`, `bxp()`, and hence implicitly `boxplot()`. All except the former also gain a new `panel.last`. Suggested by Marieke Stolte and Julian Welz.
- `plot.default()` gets a new switch `lim2` to indicate that the default `xlim` and `ylim` range computations only use *jointly* finite x- and y-values; indirectly suggested in PR#7986 by Jincai Jiang.
- '`Rman.css`', the stylesheet for the `texi2any`-generated HTML versions of the R manuals, was improved following joint efforts at the 'R Dev Days' of useR! 2024 and RSECon25.

- New system Rd macro `\manual{`*name*`}{`*node*`}` to refer to a specific section *etc* (possibly empty, for '`Top`') of one of the R manuals (e.g., '`R-exts`').
- `list.files()` now has optional argument `fixed` to allow literal pattern matching. Suggested and implemented by Duncan Murdoch in PR#18925.
- `substring(txt, i1, i2)`, `substr(..)` and their replacement versions now accept `i2 = NULL` to mean "to the end_of_string", thus fulfilling Kevin Ushey (and others') suggestions in PR#18851.
- The default method for `pretty()` gets a new switch `bounds` for completeness, corresponding to `.pretty()`. Also, it now catches typos such as `eps.corrected = 2`, noted in PR#18521 by Mikko Korpela.
- Plain-text help now renders Rd `\describe` lists using line breaks after item labels. Set `tools::Rd2txt_options(descStyle=NULL)` to get the old behaviour with a simple space; a `"colon"` variant is also available.
- `x %notin% table` newly in **base** is an idiom for `!(x %in% table)` and provided almost entirely for convenience and code readability, from an R-devel suggestion, after many years of private definitions mostly hidden in packages, including in R's **tools** package.
- The default method of `sequence()` gets a new option `recycle` to swap between current (not fully recycling) and future behaviour where all three of `(nvec, from, by)` are jointly recycled.
- `ftable()`s default method and the `as.table()` method for `"ftable"`s get a `perm` argument with back compatible default. The `as.table()` method also gets a `named.dim = FALSE` switch, no longer creating a named `dim(<result>)` by default.
- `match.arg()` gets a new option `several.ok = "all"`, fulfilling the wish of Jim Hester in PR#16659.
- `provideDimnames()` gets new switch `use.names`, as proposed including patch by Mikael Jagan, in PR#18975.
- The default `summary()` is more informative for character vectors, optionally resorting to the factor method via the new argument `character.method = "factor"`, e.g., when summarizing a data frame (PR#16750).
  It also provides more useful summaries for complex vectors, with `polar` determining if polar or cartesian coordinates are more relevant; mostly based on a proposal by Mikael Jagan.
- Error messages from failed Rd `\Sexpr` code now include the call from the original condition (if available). Warning messages now include Rd source locations.
- When the environment variable `R_WARN_BOUNDS_OPT` is set to `"yes"`, `options(warn = val)` will check if `val` is inside `-1:2`; this may happen unconditionally in the future.
- `weighted.residuals()` now returns weighted working residuals for `"glm"` objects. It used to return deviance residuals, but these do not give the one-step approximated `dfbeta` values, as pointed out by Ravi Varadhan. `influence.glm()` has been amended so that it now returns an object with both a `"wt.res"` and a `"dev.res"` element.
- Several influence measures for `"glm"` objects have been updated to use Pearson residuals rather than deviance residuals (the latter are not unbiased) and to not use leave-one-out estimates of dispersion in models with fixed dispersion. Thanks to Ravi Varadhan for discussions and testing of these updates.
- R now supports custom binary types which are of the form `"<system>.binary.<build>"` where `<system>` is lower-case name (letters only) of the system and `<build>` is the name of the build (alphanumeric and dashes). The corresponding paths returned by `contrib.url` (which govern the repository layout) with the above `type=` have the form `bin/<system>/<build>/contrib/<x.y>` where

<x.y> is the R version without patch level. This is a generalisation of the binary types previously supported on macOS. In addition, binary packages can have the extensions `.tar.bz2`, `.tar.xz`, `.tar.zst` and `.tar.zstd` with the corresponding compression support.

- The value of `.Platform$pkgType` which defines the native binary type for the current build of R can be overridden with `R_PLATFORM_PKGTYPE` environment variable (previously this was only set by the macOS builds of R). This should be typically only done by the author of a binary R distribution to define the type of binaries supported by that build of R.
- If R uses custom binary type in its `.Platform$pkgType` then `R CMD INSTALL --build` will produce `<package>_<ver>_R_<system>-<build>.tar.xz` binary files. Please note that future versions may use `.zstd` compression by default so package tooling should support any of the above mentioned extensions.
- New facilities have been introduced for package integrity verification and package signing. The optional file '`SHA256`' in a package contains SHA-256 hashes, lengths and paths of files provided in the package allowing for more reliable integrity verification (relative to '`MD5`'). This file is now generated by default in `R CMD INSTALL --build` and optionally via `R CMD build --sha256`. A package can be signed by creating a detached GnuPG signature file '`SHA256.sig`' signing the entirety of the '`SHA256`' file. Upon installation, the signature is verified using GnuPG, thus ensuring the integrity of the '`SHA256`' file and by induction all files in the package. Both source and binary packages can be signed. Currently, this only applies to tar-based packages, and the command-line GnuPG `gpg` tool (or equivalent tool named in the `GPG` environment variable) has to be available for signing and verification to work. `R CMD INSTALL --sign` can be used to sign binary packages, including its use with '`--build`'. System-wide GnuPG keyrings for the R installation can be placed in *R_HOME*`/etc/keyrings` as files with '`.gpg`' extensions and will be checked first before consulting user's GnuPG keyring (if present).
- Experimentally, several math library ('`math.h`') functions, such as `exp(.)`, `log1p()`, `sin()`, `atanh()`, when called from R, are now ensured to give fully accurate results in a small neighbourhood of special values, such as `exp(0) == 1`, `log1p(0) == 0`, etc. This makes R slightly less platform dependent for these functions, notably on platforms with less accurate system math libraries, see also https://blog.r-project.org/2026/01/30/debugging-sensitivity-to-c-math-library-on-linux/.
- Function `free1way()` implementing semiparametrically efficient inference procedures (likelihood- and permutation-based) in distribution-free stratified K-sample one-way layouts was added to the **stats** package. Approximate (via `power.free1way.test()`) and simulation-based (utilising `rfree1way()` for random number generation) power-assessment and sample-size planning as well as model diagnostics (available from a `plot` method and probability-probability plots in `ppplot()`) complement the functionality. Provided by Torsten Hothorn.
- `plot.data.frame()` now supports a `formula` argument.
- New Rd macro `\linkS4methods{}`, e.g., for easier linking to method docs in other packages, proposed and coded by Mikael Jagan in PR#18998.
- Class `"difftime"` is now also among the S4 `"OldClasses"`.
- New function `bitstring()` as a user-friendly wrapper around `intToBits()`, etc.
- `plot.lm()` now omits the smoother in the residuals vs leverage plot. This never made much sense, but if you really want it, you can force the old behaviour by setting `panel.raw=panel.smooth`.
- `hatvalues()` now has a method for `nls` (nonlinear model) results, prompted by the proposal of Patrice Kiener and Heather Turner in PR#18897.

- The Sweave driver `RweaveLatex` now allows the value of logical and numerical chunk options (only) to be the name of an object defined in earlier, evaluated code chunks. This is useful to pass computed values to options. For options set this way for weaving purposes, the `Rtangle` driver rather uses their default values, with a warning. By Vincent Goulet.
- The Sweave driver `RweaveLatex` gains an option to include unparsable R code or arbitrary text in code chunks. Chunks marked as `ignore.on.weave = TRUE` are completely ignored on weaving, i.e., not even parsed. The chunks however remain written out verbatim on tangling.
  Similarly, the driver `Rtangle` gains an option to control the extraction of code chunks. Chunks marked as `ignore.on.tangle=TRUE` are processed as usual on weaving, but left out of the tangled scripts (or thrown away when `split=TRUE`).
  The drivers also support the shorter aliases `weave` for `!ignore.on.weave`, and `tangle` for `!ignore.on.tangle`. A further option `ignore`, available for both drivers, sets `ignore.on.weave` and `ignore.on.tangle` at once.
  By Vincent Goulet.
- The Sweave driver `Rtangle` now provides an option `chunk.sep` to control the separator between code chunks in tangled scripts (with an unchanged default of `\n\n`). Using `check.sep=FALSE` omits the separator (between two chunks when used as a chunk option). By Vincent Goulet.
- The Sweave driver `Rtangle` now provides an option `extension` to specify the extension, without the leading dot, for the file name of a tangled code chunk when splitting is selected (the option has no effect with `split=FALSE`). If `TRUE`, the default extension of `Rtangle` (usually '`.R`') is used; if `FALSE`, no extension is added to the file name. By Vincent Goulet.
- New function `SweaveGetSourceName` in package **utils** to obtain the name of the file being woven by an `Sweave` process launched from the command line. By Vincent Goulet.

Graphics:

- The graphics engine version, `R_GE_version`, has been bumped to `17` and so packages that provide graphics devices should be reinstalled.
  The `glyphFont()` function gains a `variations` argument, which allows variable font axes to be specified, e.g., `c(wght = 100)`.
  Only the `quartz()` device and Cairo-based graphics devices (but not `cairo_pdf()`) support variable font axes so far.

Installation on a Unix-Alike:

- `configure` appends `bsdtar` to the list of command names it tries when finding a default for `TAR`.
- Failures in building the documentation for the base packages, including from parsing their Rd files, are no longer ignored by `make docs`.
- Building the R manuals now requires Texinfo 6.8 or later.
- A new `LIBR_LDFLAGS` defaulting to `LDFLAGS` allows more flexibility, e.g., for distributions such as Gentoo, Debian or Ubuntu, fixing PR#18992; additionally a `make` failure to create '`libR.pc`' should no longer be ignored, fixing PR#18993; both thanks to Benjamin Drung.

Checking an Installation:

- There is a new directory '`tests/C++Standards`' which contains small packages which can be test-installed to check support for various C++ standards. This is principally for use on Windows as on a Unix-alike this partially duplicates tests done by

`configure` — although it can also be used to test the behaviour of `R CMD INSTALL` on unsupported standards.

Package Installation:

- Specifying C++11 or C++14 in '`src/Makevars`' has long given a note: in 99% of CRAN packages the specification was unnecessary as the default C++17 sufficed (in a few cases with a deprecation warning from some compilers).
  Support for these standards has been removed: the default C++ standard will be used.
  `R CMD config` variables `CXX11`, `CXX14` and their associated `CXXxxFLAGS`, `CXXxxPICFLAGS` `CXXxxSTD`, `SHLIBCXXxxLD` and `SHLIBCXXxxLDFLAGS` variables are no longer supported and reported as 'defunct'.
- When installing binary packages the time stamp in the `"Built"` field is considered even if the package version is identical to allow binary package repositories to update binaries without the need to increase the package version.
- The `"Built"` column returned by `installed.packages()` and `old.packages()` now contains the full value of the `"Built"` entry from the (installed) '`DESCRIPTION`' file. Previously, only the R version from the `"Built"` entry was returned.
- If the timestamp passed as '`--built-timestamp`' to `R CMD INSTALL` is in RFC-2822 form (with time-zone offset) it will be converted to ISO UTC form to conform to the format used by R for purposes described above. (PR#19037, thanks to Dirk Eddelbuettel)

C-Level Facilities:

- New functions `GEcreateDD()` and `GEfreeDD()` for allocating (and initialising) `DevDesc` structures. Can be used by external graphics devices. Satisfies PR#18292.
- New function `R_eval_with_gd()` designed to "lock" a graphics device during call to `eval()` within device drawing routines. External *on-screen* graphics devices should consider using this function to avoid crashes from the device being closed by the user (e.g., a mouse click) during drawing.
- Declarations for non-API functions `LOGICAL0`, `INTEGER0`, `REAL0`, `COMPLEX0`, and `RAW0` have been removed from installed header files.
- Non-API entry points `Rf_acopy_string` and `Rf_lazy_duplicate` have been added to those reported by `R CMD check`.
- Packages using the non-API functions `ATTRIB` and `SET_ATTRIB` will now receive check '`NOTE`'s. See 'Writing R Extensions' for alternatives to use.
- `R CMD check` '`NOTE`'s on the use of these non-API entry points have been upgraded to '`WARNING`'s in preparation for removing declarations and, where possible, hiding these entry points: `R_nchar`, `R_tryWrap`, `Rf_GetOption`, `R_lsInternal`, `BODY`, `FORMALS`, `CLOENV`, `SET_TYPEOF`, `STRING_PTR`, `R_duplicate_attr`, `getConnection`, `R_data_class`, `STRING_PTR`, `SET_OBJECT`, `ATTRIB`, `SET_ATTRIB`, `Rf_findVarInFrame3`.
- The deprecated macros `CHARACTER_DATA` and `CHARACTER_POINTER` now return `const` pointers since using writable pointers in packages is not safe.
- New function `DATAPTR_RW()` for use in implementing ALTREP `Dataptr` methods. This function should not be used in any other contexts.
- New function `R_class()` as the `C` equivalent of `class()` in R.
- New function `Rf_isScalarString`.
- The `R_ObjectTable` type definitions formerly in '`R_ext/Callbacks.h`' are now available in the new header file '`R_ext/ObjectTable.h`'. This new header file should be used in packages instead of '`R_ext/Callbacks.h`'.

- Task callback support declared in '`R_ext/Callbacks.h`' has been enabled again. It was disabled for R 4.5.0.
- The non-API header file '`R_ext/PrtUtil.h`' is no longer copied to the installed '`include`' directory.
- New function `R_mapAttrib` for iterating over the attributes of an object.
- New functions `R_altrep_class_name` and `R_altrep_class_package` for retrieving the class name symbol and package name symbol of an ALTREP object. These should be used instead of `ALTREP_CLASS`.
- New function `R_getAttributes` returns the same result as the R function `attributes`. New functions `R_getAttribCount`, `R_getAttribNames`, `R_hasAttrib`, `R_nrow`, and `R_ncol`.
- The function `Rf_isFrame` has been removed; use `Rf_isDataFrame` instead.
- The function `VECTOR_PTR` has been removed.
  The non-API function `DATAPTR` is no longer declared in an installed header file.
- These entry points are now marked as hidden and no longer have declarations in installed header files: `ENVFLAGS`, `EXTPTR_PROT`, `FRAME`, `ENCLOS`, `EXTPTR_PTR`, `EXTPTR_TAG`, `HASHTAB`, `IS_S4_OBJECT`, `LEVELS`, `NAMED`, `OBJECT`, `R_shallow_duplicate_attr`, `Rf_isValidString`, `Rf_lazy_duplicate`, `Rf_NonNullStringMatch`, `SETLENGTH`, `SETLEVELS`, `SET_BODY`, `SET_CLOENV`, `SET_ENCLOS`, `SET_FORMALS`, `SET_ENVFLAGS`, `SET_FRAME`, `SET_GROWABLE_BIT`, `SET_HASHTAB`, `SET_NAMED`, `SET_S4_OBJECT`, `SET_TRUELENGTH`, `STDVEC_DATAPTR`, `TRUELENGTH`, `UNSET_S4_OBJECT`, `XTRUELENGTH`.
- New experimental API for working with variable bindings in environments. The function `R_GetBindingType` identifies the binding type without forcing a delayed or active binding. New bindings care created by `R_MakeDelayedBinding`, `R_MakeForcedBinding` and `R_MakeMissingBinding`. Delayed and forced bindings can be inspected with `R_ForcedBindingExpression`, `R_DelayedBindingExpression`, and `R_DelayedBindingEnvironment`. Bindings for `...` in function call environments can be examined with `R_findDotsEnv`, `R_DotsExist`, `R_DotsLength`, `R_DotsNames`, `R_DotsElt`, `R_GetDotType`, `R_DotDelayedExpression`, `R_DotDelayedEnvironment`, and `R_DotForcedExpression`. Contributed by Lionel Henry, Davis Vaughan, and Charlie Gao in PR#18928.
- New function `R_envSymbols` returns a vector of the symbols for which an environment contains bindings (PR#18928).
- `R CMD check` '`NOTE`'s on the use of the non-API entry points `PRCODE`, `SET_PRCODE`, `PRENV`, `SET_PRENV`, `PRVALUE`, `SET_PRVALUE`, `R_PromiseExpr`, and `Rf_allocSExp` have been upgraded to '`WARNING`'s in preparation for removing declarations and, where possible, hiding these entry points.
- New functions `R_getVar` and `R_getVarEx`. In most cases these can be used instead of the non-API functions `Rf_findVar` and `Rf_findVarInFrame`. Other cases can be handled by the new API for working with variable bindings.
- Packages using any non-API variables will now receive check '`NOTE`'s.
- New function `R_getRegisteredNamespace` finds and returns the namespace environment registered for a name, or `R_NilValue` if none is found.

Utilities:

- `R CMD check` now reports further `clang` warnings including `-Wkeyword-macro`. This is most commonly seen for the C23 keywords `bool`, `true` and `false`, but is also seen for `nullptr` (C23 and C++11). Masking these by macros *may* generate correct code (but not always) and always results in confusing-to-read code.

- The cascaded stylesheet '`R.css`' now adds less vertical white space when rendering Rd files' `\itemize`, `\enumerate`, and `\describe` list `\item`s to HTML.
  Where R is used via extensive IDEs such as `RStudio`, their maintainers may need to adapt '`R.css`' files accordingly.
- `R CMD check` when passed a tarball looks at its contents and so no longer derives the package name from the tarball name (which can now be arbitrary).
- Package '`README.md`' files are now installed and featured in HTML help.
- `package.skeleton()` now invisibly returns the path of the created directory, similar to `prompt()` et al., and allows creating blank skeletons without any R objects.

Deprecated and Defunct:

- `structure(NULL, <name> = <val>)` is now defunct.
- Three `tk*.slaves()` functions were deprecated in favor of `tk*.child()` which do need Tcl/Tk version `8.6` or higher.

Bug Fixes:

- `chol(x, pivot=TRUE)` now zeroes the trailing submatrix when the algorithm stops early. Following the help page instructions on how to reconstruct the matrix in positive semidefinite cases gave incorrect results if the rank was 2 or more less than the dimension (PR#15853).
- Profiling `"mle"` objects (**stats4** package) could fail because profile MLE used starting value from full MLE. Now starts from profile MLE from previous step.
- Setting `attributes` on primitive functions is now an error; previously it modified without copying (as noticed by Henrik Bengtsson on the R-devel mailing list).
  Relatedly, changing the `environment()` of a primitive function does no longer happen and gives a warning (to become an error, later).
- Dataset `LifeCycleSavings` had a misspelled row name: `"Guatamala"` was corrected to `"Guatemala"`.
- The `Pair(x,y)` construction in the formula interface to paired tests did not work with the `subset` argument, due to the absence of a `[.Pair` method, which has now been added (PR#18898).
- Subsetting `"table"` objects keeps S3 classes more consistently, fixing PR#18845.
- When `browser()` is active or when otherwise debugging, R code such as `if(TRUE)` no longer triggers a new browser level, fixing PR#18885 (differently than the fix for PR#15770), with thanks to Ivan Krylov.
- `text()` now truncates `labels` to maximum length of `x` and `y` (if it is longer), fixing PR#7084. Thanks to Heather Turner, Ella Kaye, and Philippe Grosjean.
- `<Date> %in% set` is again as fast as it was before R 4.3.0, *via* a new S3 method `mtfrm.Date`.
  `<character> %in% <Date>` and vice versa are now documented to work in concordance with `==` and as an exception to the typical `match()` behaviour which relies on "univariate" `mtfrm()` alone.
- It is now possible to convert *to* `"snpc"` units via, e.g., `grid::convertWidth()`, fixing PR#18915. Thanks to Trevor Davis.
- The `precip` dataset had typos in the names `"Bismarck"` and `"Pittsburgh"` (PR#18895).
- `tools::checkReplaceFuns()` now deals better with replacement methods *not* available as regular functions in the namespace.
- `length(xpr) <- n` now also works for `expression` vectors, the same as for `list`s, i.e., padding with `NULL` entries if the length is increased. Thanks to Mikael Jagan's PR#18917.

- `<complex>[i] <- NA` now only sets the *real* part to `NA`, consistently with the `as.complex(NA)` behaviour since R 4.4.0; thanks to Mikael Jagan's PR#18918.
- `format(<list>)` calling `format.default()` now calls generic `format()` on each list element (rather than `format.default()`).
- `tools::Rd2HTML()` detects paragraphs in more cases of `\Sexpr`-generated Rd code, fixing a regression in R 3.6.0 originally reported by Georgi Boshnakov and later also in PR#18470. An inactive `#ifdef` block no longer breaks the paragraph.
- Parsing a multi-line string from Rd `\Sexpr` code no longer doubles the newlines.
- `requireNamespace(.., versionCheck = *)` now shows an error and returns `FALSE` when the package version check is not fulfilled, thanks to Iris Simmons and Trevor Davis's PR#18255. Additionally, the `versionCheck` argument is checked and when invalid, an error is signalled to avoid subsequent confusing error messages.
- `requireNamespace()` now throws an error even if the name space is already loaded if it is called with an unmatched argument such as `quitely = TRUE` or `quiet = TRUE` (argument `quietly` cannot be abbreviated as it follows `...`).
- `rep()` signals more specific error messages, notably when using both `times` and `each` optional arguments. Fixes PR#18926, thanks to Ella Kaye and collaborators at useR! 2025's 'R Dev Day'.
- `jitter(x, amount, factor)` becomes more robust for non-finite `x` or also negative `amount`, `factor`. `help(jitter)` is easier to read, fixing PR#17279, thanks to its contributors.
- Plain-text help (`Rd2txt`) lacked markup for '`\cite`'d titles, which are now wrapped in single quotes.
- Default column labels are now correctly aligned when printing a matrix with right alignment (PR#18943, thanks to Mikael Jagan).
- `poly(fct, n)` signals an error again, for a non-`ordered` factor `fct`, thanks to Roland Fuss' notice on R-devel.
- `terms(<formula>, simplify=TRUE)` now works correctly also in the cases where the formula has more than one term with NAs such as `f(u, b=NA)` or `g(NA)`, finally fixing PR#15275 thanks to Mikael Jagan's patch.
- The help page `example(matplot)` now correctly labels Sepal vs Petal dimensions in its first iris data plot, thanks to Jeff Dick's post on R-devel.
- `diff(m, lag, dif)` for matrix `m` now still returns matrices, also when `lag * dif > nrow(m)` (PR#18972, thanks to Mikael Jagan and Suharto Anggono). `diff(<ts-matrix>)` remains matrix, even when it has length zero.
- Values returned by active binding functions are now marked as not mutable to prevent unintended mutation in complex assignment operations.
- The default method of `sequence()` now allows to fully recycle all three of `(nvec, from, by)` as always documented; with thanks to Mikael Jagan's report and patch in PR#18304. Its behaviour remains back compatible for now, but can be switched to future behaviour via environment variable `R_sequence_recycle`, or explicit `recycle = TRUE`.
- Subassignment, `x[ind] <- val`, to 1-dimensional `array`s no longer "drops" them to simple vectors when `ind` is a name, fixing PR#18973, reported by Thomas Soeiro.
- The `implicitGeneric` definition for `toeplitz()` produced a default method that failed when `toeplitz(x, r)` was called with more than the first argument. Simple fix from Mikael Jagan.
- The error message for `array(dim = c())` now uses `"'dim'"`, thanks to Hugo Gruson.
- Some legal but strange nested complex assignments are handled a bit more robustly in interpreted code.

- `cacheMetaData()` does not to mangle primitive generics.
- The S4 method `norm(<ANY>, type = <missing>)` for the implicit generic `norm()` now uses `type = "O"` as has been the intention for more than ten years, fixing PR#18979 thanks to Mikael Jagan.
- On Windows terminals *`Ctrl-C`* now breaks out of waiting for keyboard input, fixing PR#18982.
- The default `summary()` and its factor method now label missing values '`NAs`' rather than '`NA's`', fixing a decades-old grammatical faux pas inherited from S (PR#16348 and PR#18948).
- Asymmetric `toeplitz(x, r)` is type consistent now, even when `r` of length 0 or 1 is not used directly. Fixes PR#18996, thanks to Mikael Jagan's patch and test.
- `format(r)` now keeps `names(r)` also for raw objects `r`.
- `rep.int()` and `rep_len()` again drop names also for factors, fixing a regression in R 4.0.0. (Thanks to a report by Rui Barradas and PR#18999 by Suharto Anggono.)
- `str(x)` with a `list`-like `x` of a class having an `str()` S3 method using `NextMethod()`, now no longer outputs `"List of <...>"` in the default method, thanks to Ian Farm's report and fix proposal in PR#19001.
- `as.POSIXlt(ch)`, `as.Date(ch)` etc now also work correctly when `ch` contains `"Inf"`, via `strptime(.)`, fixing PR#19006.
- `c(<POSIXlt>, ..)` now fully preserves sub-second accuracy, fulfilling a part of Suharto Anggono's wishlist PR#18989.
- When subassigning `"POSIXlt"`, e.g., `<dat>[i] <- v`, `v` is now converted to the time zone of `dat`, before being assigned, see PR#18919.
- The `"tzone"` attribute of `"POSIXlt"` objects should no longer depend on earlier calls, fixing PR#19005 thanks to Ivan Krylov, a patch by Suharto Anggono and reports of "Non-deterministic POSIXlt tzone in Europe/London .." on R-devel.
- In intervals with very low density `cdplot()` was unreliable, returning probabilities that are greater than one or `NaN` (PR#19011). Fix from Christoph Dalitz and Achim Zeileis.
- `R CMD build` now installs packages providing their own vignette engine (PR#18191). Patch by Paul Smith.
- `ar.burg()`, `ar.yw()` and `ar.ols()` deal better with non finite AIC values thanks to Simone Giannerini's remarks on R-devel.
- Parsing deeply nested expressions no longer should segfault in certain circumstances, fixing five different checks on context stack size, thanks to Kevin Ushey's report and Michael Chirico's patch in PR#18994.
- The internal `cwilcox` function used in computations for the Wilcoxon distribution is now based on an implementation of a formula by Andreas Löffler provided by Andreas Löffler, Aidan Lakshman, and Ivan Krylov, substantially reducing the memory complexity of the original implementation (PR#18655).
- When `Ops` group methods for matrix-`"ts"` return a zero-extent matrix, its type is correct now. E.g., `<int> + <dbl>` now returns type `"double"`, see also PR#18972.
- `intToUtf8(<negative>)` now returns `NA` and no longer accesses invalid memory, thanks to Kevin Ushey's report with patch in PR#19052.
- When `file.choose()` is terminated with an `EOF`, it now signals an error rather than returning garbage, thanks to Kevin Ushey's report and patch in PR#19053,



Changes in R 4.5.3

Utilities:

- `tools/fetch-recommended` can be used instead of `tools/rsync-recommended` to fetch recommended packages into R sources using `curl` on systems without `rsync` or behind firewalls.

Package Installation:

- C++ standard specifications (`CXX_STD =` in '`src/Makevars*`' and in the `SystemRequirements` field of the '`DESCRIPTION`' file) are now checked more thoroughly. Invalid values are still ignored but now give a warning, as do contradictory specifications.
- (Preliminary) support for C++26 has been extended to Windows.
- A non-zero exit status from '`cleanup`', '`cleanup.win`' or '`cleanup.ucrt`', if requested via options '`--clean`' or '`--preclean`', is now reported with a warning.

Bug Fixes:

- `all.equal(obj, simple, check.class=FALSE)` now is true, also when `simple` is a bare atomic vector and `obj` has a simple class, fixing the first part of PR#18971 thanks to Jan Gorecki.
- `str(x, give.attr=FALSE)` no longer shows attributes when `x` is a zero length `"Date"` or `"POSIXt"` object.
- Tweaks to binning for `bw.SJ()` and `bw.ucv()` in the very rare case of data which have an extremely small range compared to their absolute values, e.g., `x <- 6e9 + 6:9`.
- Formatting book-type `bibentry` objects now converts LaTeX accents also in the publisher and series fields.
- `model.frame(~1, list(), na.action=na.pass)` and similar "border-line" uses no longer produce invalid data frames, fixing PR#18977, reported with patch by Mikael Jagan.
- `length(<POSIXlt>) <- v` is more careful about balancing, notably when `v` is not integer, thanks to Suharto Anggono's remarks on the mailing list R-devel.
- `approx(<x_with_ties>, <y_with_NA>, na.rm = FALSE)` now should always call the `ties()` function, fixing PR#17604 reported by Bill Dunlap.
- `besselJ(1, 1e-15)` and similar now give correct results, thanks to Leo Mada and other "R-help"ers.
- `vignette(`*pkg*`::`*topic*`)` is now a documented usage variant and confines vignette retrieval to the specified package.
- `pretty(ch)` again correctly works with `ch` a character vector of numbers.
- `persp()` labels the three axes correctly also when C level `atan2pi()` is available, fixing PR#19007 by Klaus Schliep.
- Large (tall) subscripts in plotmath expressions are now positioned correctly. They were being positioned too high.
- `Ops` group (`Arith`, `Compare`, ...) methods for matrix-`"ts"` now do return (zero length) matrices even when the two series do not overlap; a very partial fix of PR#18972, which is to be fully addressed later.



Changes in R 4.5.2

Utilities:

- `R CMD check` now handles archives with extension '`.tar`' or '`.tar.zstd`' (where `zstd` compression is supported by the R build).

Bug Fixes:

- `t.test(c(1:3, Inf))` and similar no longer produce an error but return a (still not so useful) `"htest"` result, fixing PR#18901, thanks to Jesse Alderliesten.
- `attr(., "tsp") <- val` now uses `getOption("ts.eps")` instead of hardwired `1e-5`; consequently, `ts(.., ts.eps=*)` now passes `ts.eps` to the `"tsp"` setting C code; both fixing a long-standing 'FIXME'.
- `insertSource()` now ignores the internal `.packageName` object, avoiding a superfluous message.
- In static HTML help, links to vignette files from the default '`doc/index.html`' page now also work for packages not installed in the default `.Library`, thanks to a report by Patrice Kiener.
- `fixInNamespace("`*S3method*`")` failed to update the S3 methods table when the generic was not on the search path.
- In plain-text help (`Rd2txt`), an initial newline from an Rd inline `\eqn` no longer breaks the paragraph.
- `hist(*, log = "x")` now works without a warning, thanks to Martin Smith's PR#18921.
- Subassigning `"POSIXlt"`, i.e., `<tdat>[i] <- val` and `<tdat>[[i]] <- val` now rebalance as they should; thanks mostly to Mikael Jagan's PR#18919.
- `<POSIXlt>[*]` (re-)setting `"balanced"`, fixing PR#18681 comment #7, thanks to Mikael Jagan.
- All four of `{col,row}{Sums,Means}(Z, na.rm=TRUE)` now correctly work with complex `Z` where `is.na(Re(Z))` differs from `is.na(Im(Z))`, fixing PR#18942, unearthed by Dirk Eddelbuettel.
- Fix for glyph rendering on the `quartz()` device when there is other ("normal") text drawn on the device. The problem was that the text transformation matrix was not reset so glyphs would be rendered incorrectly (often completely outside the device, i.e., not visible).
- Functions `install.packages()` and `download.packages()` again consult option `download.file.method` when the download method is unspecified.
- Tanguy Barthelemy and colleagues at the 'R Dev Day' following Rencontres R in May 2025 extended the help page of `lm()`, fixing PR#18058. As suggested by Thomas Soeiro, such notes were also added to the `glm()`, `poly()` and `splines::bs()` and `ns()` pages.
- `lbeta(1i, 1)` now signals an error, as `lbeta()` is not implemented for `complex`, fixing PR#18946 from Ben Bolker and Kasper Kristensen.
- The Cairo-based SVG device uses `pt` as the default document unit also with Cairo >= 1.17.8 (PR#18912).
- `pretty(*, eps.correct = 2)` has been fixed, e.g., to avoid over 1 million length result for `pretty(c(0, 1e-322), eps.correct = 2)`.
- When `"POSIXlt"` date-time objects are `NA`-padded from subsetting or increasing `length` in the `` `[` `` or `` `length<-` `` methods, the `"balanced"` attribute is set to `NA`, now; noted in PR#18681, thanks to Mikael Jagan.



Changes in R 4.5.1

New Features:

- The internal method of `unzip()` now follows `unzip 6.00` in how it handles extracted file paths which contain `"../"`. With thanks to Ivan Krylov.

Installation:

- Standalone `nmath` can be built with early-2025 versions of `clang`-based compilers such as LLVM `clang` 20, Apple `clang` 17 and Intel `icx` 2025.0.
- Tcl/Tk 9 can be used to build package `tcltk`: this has become the default in some Linux distributions.

Bug Fixes:

- Java detection in `javareconf` could not detect '`libjvm.*`' in the `zero` variant of the JDK (PR#18884). All valid variants as of JDK 24u are now supported.
- `factanal(.., rotation=*)` now correctly updates `rotmat`, fixing PR#18886.
- `dnbinom(<large>, <muchlarger>, ..)` now is `0` correctly, instead of `NaN` or `Inf` sometimes.
- `dbinom(<large>, n=Inf, ..)` is `0` now correctly, instead of `NaN` which also fixes many `dnbinom()` cases, notably those mentioned in PR#16727 comment #5.
- Fixing C level "binomial deviance" `bd0()` for extreme arguments (preventing under-/overflow) solves more PR#16727 cases and also prevents some full accuracy loss in such cases for `dbinom()`, `dnbinom()`, and via `dbinom_raw()` potentially `dgeom()`, `dhyper()`, `dbeta()`, and `df()`.
- `signif(1.**e308, digits)` no longer truncates unnecessarily (but still to prevent overflow to `Inf`), fixing PR#18889.
- `prettyNum(*, zero.print={>=1-char}, replace.zero=TRUE)` now works as documented, thanks to Marttila Mikko and Ivan Krylov's messages on R-devel.
- `pbeta(x, a,b, ..)` for very large `a,b` no longer returns `NaN` but the correct values (0 or 1, or their logs for `log.p = TRUE`). This improves Mathlib's C level `bratio()` and hence also `pnbinom()`, etc..

Changes in R 4.5.0

New Features:

- `as.integer(rl)` and hence `as.raw(rl)` now work for a `list` of `raw(1)` elements, as proposed by Michael Chirico's PR#18696.
- **graphics**' `grid()` gains optional argument `nintLog`.
- New functions `check_package_urls()` and `check_package_dois()` in package **tools** for checking URLs and DOIs in package sources.
- New `head()` and `tail()` methods for class `"ts"` time series, proposed by Spencer Graves on R-devel.
- New `qr.influence()` function, a "bare bones" interface to the `lm.influence()` leave-one-out diagnostics computations; wished for in PR#18739.
- Package `citation()` results auto-generated from the package metadata now also provide package DOIs for CRAN and Bioconductor packages.
- New function `grepv()` identical to `grep()` except for the default `value = TRUE`.
- `methods(<pkg>:::<genfun>)` now does report methods when neither the generic nor the methods have been exported.
- `pdf()` gains an `author` argument to set the corresponding metadata field, and logical arguments `timestamp` and `producer` to optionally omit the respective metadata. (Thanks to Edzer Pebesma.)
- `grDevices::glyphInfo()` gains a `rot` argument to allow per-glyph rotation. (Thanks to Daniel Sabanes Bove.)

- Package **tools** now exports functions `CRAN_current_db()`, `CRAN_aliases_db()`, `CRAN_rdxrefs_db()`, `CRAN_archive_db()`, and `CRAN_authors_db()`.
- Package **tools** now exports functions `R()` and `parse_URI_reference()`.
- Package **tools** now exports functions `base_aliases_db()` and `base_rdxrefs_db()`.
- It is now possible to set the background color for row and column names in the data editor on Windows (`Rgui`).
- `Rterm` on Windows now accepts input lines of unlimited length.
- `file.info()` on Windows now provides file owner name and domain.
- `Sys.info()` on Windows now provides current user domain.
- `findInterval()` gets new arguments `checkSorted` and `checkNA` which allow skipping relatively costly checks; related to PR#16567.
- `pnorm(x)` underflows more gracefully.
- `get(nam, env)` now signals a *classed* error, `"getMissingError"`, as "subclass" of `"missingArgError"` where the latter is used also in similar situations, e.g., `f <- function(x) exp(x); try(f())` .
- The set operations now avoid the `as.vector()` transformation for same-kind apparently vector-like operands.
- `md5sum()` can be used to compute an MD5 hash of a raw vector of bytes by using the `bytes=` argument instead of `files=`. The two arguments are mutually exclusive.
- Added function `sha256sum()` in package **tools** analogous to `md5sum()` implementing the SHA-256 hashing algorithm.
- The `xtfrm()` method for class `"AsIs"` is now considerably faster thanks to a patch provided by Ivan Krylov.
- The `merge()` method for data frames will no longer convert row names used for indexing using `I()`, which will lead to faster execution in cases where `sort = TRUE` and `all.x` and/or `all.y` are set to `TRUE`.
- The **methods** package internal function `.requirePackage()` now calls `requireNamespace(p)` instead of `require(p)`, hence no longer adding packages to the `search()` path in cases methods or class definitions are needed. Consequently, previous workflows relying on the old behaviour will have to be amended by adding corresponding `library(p)` calls.
- More R-level messages use a common format containing `"character string"` for more consistency and less translation work.
- `available.packages()` and `install.packages()` get an optional switch `cache_user_dir`, somewhat experimentally.
- The `sunspot.month` data have been updated to Oct 2024; because of recalibration also historical numbers are changed, and we keep the previous data as `sunspot.m2014` for reproducibility.
- The `quartz()` device now supports alpha masks. Thanks to George Stagg, Gwynn Gebeyhu, Heather Turner, and Tomek Gieorgijewski.
- The `print()` method for date-time objects (`POSIX.t`) gets an optional `digits` argument for *fractional* seconds, passed to improved `format.POSIXlt()`; consequently, `print(<date.time>, digits = n)` allows to print fractions of seconds.
- `install.packages()` and `download.packages()` download packages simultaneously using `libcurl`, significantly reducing download times when installing or downloading multiple packages.
- Status reporting in `download.file()` has been extended to report the outcome for individual files in simultaneous downloads.
- The Rd `\link` macro now allows markup in the link text when the topic is given by the optional argument, e.g., '`\link[=gamma]{\eqn{\Gamma(x)}}`'.

- If `La_library()` is empty, `sessionInfo()` still reports `La_version()` when available.
- `seq.int(from, to, by, ....)` when $|by| = 1$ now behaves as if `by` was omitted, and hence returns `from:to`, possibly integer.
- `seq.Date(from, to, by, ....)` and `seq.POSIXt(..)` now also work when `from` is missing and sufficient further arguments are provided, thanks to Michael Chirico's report, patch proposal in PR#17672 and 'R Dev Day' contributions.
  The `Date` method also works for `seq(from, to)`, when `by` is missing and now defaults to `"1 days"`.
  It is now documented (and tested) that the `by` string may be *abbreviated* in both `seq` methods.
  Both methods return or keep internal type `"integer"` more consistently now. Also, `as.POSIXct({})` is internally integer.
- `duplicated()`, `unique()`, and `anyDuplicated()` now also work for class `expression` vectors.
- New function `use()` to use packages in R scripts with full control over what gets added to the search path. (Actually already available since R 4.4.0.)
- New connection type `zstdfile` for files compressed by `zstd` if R was built with such support. `file()` and `gzfile()` can automagically read such files.
- `memCompress()` and `memDecompress()` have options to use `zstd` compression if R was built with support for it.
- There is some support for `zstd` compression of tarballs in `tar()` and `untar()`. (This depends on OS support of `libzstd` or by `tar`.)
- `print(summary(<numbers>))` gets new optional argument `zdigits` to allow more flexible and consistent (double) rounding. The current default `zdigits = 4L` is somewhat experimental. Specifying both `digits = *, zdigits = *` allows behaviour independent of the global `digits` option.
- The `format()` method for `"difftime"` objects gets a new back compatible option `with.units`.
- A `summary()` method for `"difftime"` objects which prints nicely, similar to those for `"Date"` and `"POSIXct"`.
- `unique()`'s default method now also deals with `"difftime"` objects.
- `optimize(f, *)` when `f(x)` is not finite says more about the value in its `warning` message. It no longer replaces `-Inf` by the largest *positive* finite number.
- The documentation of `gamma()` and `is.numeric()` is more specific, thanks to the contributors of PR#18677.
- New dataset `gait` thanks to Heather Turner and Ella Kaye, used in examples.
- New datasets `penguins` and `penguins_raw` thanks to Ella Kaye, Heather Turner, and Kristen Gorman.
- `isSymmetric(<matrix>)` gains a new option `trans = "C"`; when set to non-default, it tests for "simple" symmetry of complex matrices.
- `model.frame()` produces more informative error messages in some cases when variables in the formula are not found, thanks to Ben Bolker's PR#18860.
- `selectMethod(f, ..)` now keeps the function name if the function belongs to a group generic and the method is for the generic.

BLAS and LAPACK:

- The bundled BLAS and LAPACK sources have been updated to those shipped as part of January 2025's LAPACK 3.12.1.
- It is intended that this will be the last update to BLAS and LAPACK in the R sources. Those building R from source are encouraged to use external BLAS and LAPACK and this will be required in future.

- This update was mainly bug fixes but contained a barely documented major change. The set of BLAS routines had been unchanged since 1988, so throughout R's history. This update introduced new BLAS routines `dgemmtr` and `zgemmtr` which are now used by LAPACK routines. This means that BLAS implementations are no longer interchangeable.
- To work around this, R can be configured with option '`--with-2025blas`' which arranges for the 2025 BLAS additions to be compiled into `libRlapack` (the internal LAPACK, not built if an external LAPACK is used).
  This option allows the continuation of the practice of swapping the BLAS in use by symlinking '`lib/libRblas.*`'. It has the disadvantage of using the reference BLAS version of the 2025 routines whereas an enhanced BLAS might have an optimized version (OpenBLAS does as from version 0.3.29).
- Windows builds currently use the internal LAPACK and by default the internal BLAS: notes on how to swap the latter *via* '`Rblas.dll`' are in file '`src/extra/blas/Makefile.win`'.

Installation on a Unix-Alike:

- A C23 compiler (if available) is now selected by default for compilation of R and packages. R builds can opt out *via* the `configure` flag '`--without-C23`', unless the specified or default (usually `gcc`) compiler defaults to C23: `gcc` 15 does.
  A C23 compiler is known to be selected with `gcc` 13–15, LLVM `clang` 18–20 (and 15 should), Apple `clang` 15–17 and Intel 2024.2–2025.0 (and 2022.2 should).
  Current binary distributions on macOS use Apple `clang` 14 and so do not use C23.
- The minimum `autoconf` requirement for a maintainer build has been increased to `autoconf` 2.72.
- Building the HTML and Info versions of the R manuals now requires `texi2any` from Texinfo 6.1 or later.
- Failures in building the manuals under '`doc`' now abort the installation, removing any file which caused the failure.
- Control of symbol visibility is now supported on macOS (the previous check only worked on ELF platforms).
- There is now support for installing the debug symbols for recommended packages on macOS: see '`REC_INSTALL_OPT`' in file '`config.site`'.
- `configure` is now able to find an external `libintl` on macOS (the code from an older GNU gettext distribution failed to try linking with the macOS Core Foundation framework).

Installation on Windows:

- Both building R and installing packages use the C compiler in C23 mode.
- R on Windows by default uses `pkg-config` for linking against external libraries. This makes it easier to test R and packages with alternative toolchains (such as from Msys2, e.g., testing with LLVM and possibly with sanitizers). It also allows more significant Rtools updates within a single R minor release.
- The installer scripts for Windows have been tailored to Rtools45, an update of the Rtools44 toolchain. It is based on GCC 14. The experimental support for 64-bit ARM (`aarch64`) CPUs is based on LLVM 19. R-devel and R 4.5.x are no longer maintained to be buildable using Rtools44 and it is advised to switch to Rtools45.

Deprecated and Defunct:

- `is.R()` is defunct. Environment variable `_R_DEPRECATED_IS_R_` no longer has any effect.

- Deprecated (for more than 9 years!) functions `linearizeMlist`, `listFromMlist`, and `showMlist` and the `"MethodsList"` class for S4 method handling were removed from package **methods**. Deprecated functions `balanceMethodsList`, `emptyMethodsList`, `insertMethod`, `makeMethodsList`, `mergeMethods`, `MethodsList`, `MethodsListSelect`, and `SignatureMethod` were made defunct, as were the `"MethodsList"` branches of functions `assignMethodsMetaData`, `finalDefaultMethod`, and `MethodAddCoerce`.
- `getMethods(*, table = TRUE)` is deprecated.
- Building with the bundled (and old) version of `libintl` is deprecated and now gives a `configure` warning. This should be selected only if neither the OS's `libc` (as on GNU Linux) nor an external `libintl` library provide suitable functions.
  Instead install `libintl` from a recent version of GNU gettext (available for macOS) or use `configure` option '`--disable-nls`'.
  The ability to use the bundled version may be removed as soon as R 4.5.1.
- The deprecated `xfig()` graphics device has been removed.

Package Installation:

- Packages are now installed using C23 where supported by the OS and R build.
  Packages using R's compiler settings can ask **not** to use C23 *via* including `USE_C17` in `SystemRequirements` or can be installed by `R CMD INSTALL --use-C17`. (Some packages ignore these settings in their `configure` script or when compiling in subdirectories of '`src`', as will those using a '`src/Makefile`'.)
- Source installs now report the package version in the log.
- There is preliminary support for C++26 with GCC >= 14, Apple `clang++` >= 16 and LLVM `clang++` >= 17.

C-Level Facilities:

- The non-API and hidden entry points `Rf_setIVector`, `Rf_setRVector` and `Rf_setSVector` have been removed.
- The internal code for changing the parent of an environment now signals an error if the new parent is not an environment or if the change would create a cycle in the parent chain.
- `SET_TYPEOF` now signals an error unless the old and new types have compatible memory structure and content. Use of `SET_TYPE` in package C code should be avoided and may be deprecated in the near future. It is better to allocate an object of the desired type in the first place.
- The set of LAPACK (double and complex) routines declared in the headers '`R_ext/Lapack.h`' and '`R_ext/Applic.h`' has been extended, mostly to routines actually in use by packages.
- Memory allocation messages now use the (non-SI notation) `"Mb"`, `"Gb"` , ..., and `"Mbytes"` strings as *arguments* instead of as part of the (translatable format) string. This is one step for PR#18297; from Henrik Bengtsson.
- Header '`R_ext/Constants.h`' (included by '`R.h`') now always includes header '`float.h`' or '`cfloat`' for constants such as `DBL_MAX`.
- Strict R headers are now the default. This removes the legacy definitions of `PI`, `Calloc`, `Realloc` and `Free`: use `M_PI`, `R_Calloc`, `R_Realloc` or `R_Free` instead.
- The deprecated and seemingly never-used S-compatibility macros `F77_COM` and `F77_COMDECL` have been removed from header '`R_ext/RS.h`'.
- The `enum Rboolean` defined in header '`R_ext/Boolean.h`' now has a fixed underlying type of `int` on platforms whose C compiler supports this.

This is a C23 feature (taken from C++11) and also supported in all C standards by some versions of `clang` (from LLVM and Apple) and (with a warning when using '`-pedantic`') by GCC when in C17 mode.
A fair amount of code has assumed this: it may be changed to a smaller type in future. In particular, as standard compilers do not check the validity of assignment to an `enum`, it has been possible to assign `NA_INTEGER` to an `Rboolean` variable, coerce it to `int` and recover the value.
If there were a platform which used an underlying type of a different size this would be an ABI-breaking change (but we are unaware of any such platform).

- Header '`R_ext/Boolean.h`' now ensures that a `bool` type is available either as a keyword (C23 and C++) or by including the C99 header '`stdbool.h`'. This is being used internally in R to replace `Rboolean` by `bool`.
- There are new functions `asRboolean` and `asBool`, variants of `asLogical` more suited to converting logical arguments to `Rboolean` or to `bool`. They require a length-one input and throw an error if that evaluates to `NA`.
- Header '`R_exts/Error.h`' now ensures that `Rf_error` and similar are given a `noreturn` attribute when used from C++ under all compilers.
- Header '`R_exts/Utils.h`' no longer contains a declaration for `F77_SUB(interv)`. This is intended to be called from Fortran and was wrongly declared: `LOGICAL` in Fortran corresponds to `int *` not `Rboolean *`.
- Defining `R_INCLUDE_BOOLEAN_H` to `0` before including headers '`R.h`' or '`Rinternals.h`' (or any other header which includes '`R_ext/Boolean.h`') stops the inclusion of header `R_ext/Boolean.h` which 'defines' constants `TRUE`, `FALSE`, `true`, `false` and the type `bool` which some package maintainers wish to avoid.
Note that the last three are keywords in C23 and C++11 so cannot be avoided entirely. However, with commonly-used compilers they can be masked by a macro of the same name, often with a warning.

C-Level API:

- The 'Writing R Extensions' Texinfo source now contains very experimental annotations for more clearly identifying the API status of C entry points. These annotations are used to produce indices for API, experimental API, and embedded API entry points in the rendered versions. This is very preliminary and may be dropped if a better approach emerges.
Also for Fortran-callable entry points which are part of the API.
- 'Writing R Extensions' has a new section 'Moving into C API compliance' to help package authors move away from using non-API endpoints. This section will continue to be updated as work on clarifying and tightening the C API continues.
- New API function `R_mkClosure`. This checks that its arguments are valid and should be used instead of `allocSExp(CLOSXP` followed by `SET_FORMALS`, `SET_BODY`, and `SET_CLOENV`.
- New API functions `R_ClosureFormals`, `R_ClosureBody`, and `R_ClosureEnv` for extracting closure components. The existing functions `R_ClosureExpr` and `R_BytecodeExpr` have also been added to the API.
- New API function `R_ParentEnv` corresponding to R's `parent.env()`.
- Further non-API entry points have been added to those reported by `R CMD check`: `COMPLEX0`, `ddfind`, `DDVAL`, `ENSURE_NAMEDMAX`, `ENVFLAGS`, `FRAME`, `HASHTAB`, `INTERNAL`, `IS_ASCII`, `IS_UTF8`, `LEVELS`, `NAMED`, `PRSEEN`, `RDEBUG`, `REAL0`, `Rf_findVarInFrame3`, `SET_BODY`, `SET_CLOENV`, `SET_FORMALS`, `SET_PRSEEN`, `SET_RDEBUG`, `STRING_PTR`, `SYMVALUE`, and `VECTOR_PTR`. Any declarations for these in public header files will be removed in the near future, and they will be hidden where possible.

- Some `R CMD check` 'NOTE's on the use of non-API entry points have been upgraded to 'WARNING's in preparation for removing declarations and, where possible, hiding these entry points.
- Additional non-API entry points have been added to those reported by `R CMD check`: `IS_LONG_VEC`, `PRCODE`, `PRENV`, `PRVALUE`, `R_nchar`, `Rf_NonNullStringMatch`, `R_shallow_duplicate_attr`, `Rf_StringBlank`, `SET_TYPEOF`, `TRUELENGTH`, `XLENGTH_EX`, and `XTRUELENGTH`.
- Enable defining `R_NO_REMAP_RMATH` and calling `Rf_*()` as has been documented in 'Writing R Extensions' for a while, fixing PR#18800 thanks to Mikael Jagan and Suharto Anggono.
- `R_GetCurrentSrcref(skip)` now skips calls rather than `srcref`s, consistent with counting items in the `traceback()` display. If `skip == NA_INTEGER`, it searches for the first `srcref`, starting at the current evaluation state and proceeding through the call stack; otherwise, it returns the `srcref` of the requested entry from the call stack.

Utilities:

- `R CMD INSTALL` (and hence `check`) now compile C++ code with `-DR_NO_REMAP`. 'Writing R Extensions' has been revised to describe the remapped entry points, for with the `Rf_` prefix remains optional when used from C code (but is recommended for new C code).
- `R CMD check --as-cran` notes bad parts in the 'DESCRIPTION' file's URL fields.
- `R CMD check` now reports more warnings on long-deprecated/obsolete Fortran features reported by `gfortran -Wall`. For hints on how to modernize these, see https://fortranwiki.org/fortran/show/Modernizing+Old+Fortran.
- Since almost all supported R versions now use UTF-8, `R CMD check` no longer by default reports on marked UTF-8 or Latin-1 strings in character data. Set environment variable `_R_CHECK_PACKAGE_DATASETS_SUPPRESS_NOTES_` to a false value for the previous behaviour.
- `tools::checkDocFiles()` notes more cases of usage documentation without corresponding `\alias`.
- `R CMD check` with a true value for environment variable `_R_CHECK_BASHISMS_` checks more thoroughly, including for `bash` scripts and bashisms in components of `autoconf`-generated `configure` scripts.
- `R CMD check` gains option '`--run-demo`' to check demo scripts analogously to tests. This includes a check for undeclared package dependencies: it can also be enabled separately by setting the environment variable `_R_CHECK_PACKAGES_USED_IN_DEMO_` to a true value (as done by `R CMD check --as-cran`).
- `R CMD build` now supports '`--compression=zstd`' on platforms with sufficient support for `zstd`.
- `tools::texi2pdf(..., texinputs=`*paths*`)` now *pre*pends the specified *paths* to `TEXINPUTS`. When building R itself ('`doc/NEWS.pdf`' and base vignettes) or package manuals using `R CMD Rd2pdf`, it is ensured that this R's '`Rd.sty`' takes precedence over any other (incompatible) versions in default "texmf trees".
- `tools::Rd2latex()` no longer outputs an '`\inputencoding{utf8}`' line by default; such a declaration is obsolete since LaTeX 2018-04-01.

Bug Fixes:

- `update_pkg_po()` now copies '`.mo`' files to the **translation** package even if a 'DESCRIPTION' file exists, thanks to Michael Chirico fixing PR#18694.
- Auto-generated `citation()` entries no longer include (additional) URLs in the '`note`' field (PR#18547).

- `as.data.frame.list()` gets a new option `new.names` and now preserves `NA` names, thus fixing the `format()` method for data frames, and also bug PR#18745. Relatedly, the `format()` method gets an option `cut.names`.
- `stem()` formats correctly also in cases where rounding up, e.g., from 9.96 to 10 needs more digits; thanks to Ella Kaye and Kelly Bodwin, fixing PR#8934 during 'R Dev Day' at useR!2024.
  Additionally, `stem(x)` now works normally also when `length(x) == 1`.
- **tools**' `toTitleCase()` now works better, fixing PR#18674, thanks to Shannon Pileggi, Sarah Zeller, Reiko Okamoto, and Hugo Gruson's 'R Dev Day' effort.
- Printing matrices (typically with many rows and or columns) now also omits columns when desirable according to option `max.print`, or argument `max`, respectively. This is primarily the work of Lorena Abad, Ekaterina Akimova, Hanne Oberman, Abhishek Ulayil, and Lionel Henry at the 'R Dev Day', thus fixing PR#15027.
- `Sys.setLanguage()` now warns about *some* failures to change the language.
- Printing `ls.str()` now shows `"<missing>"` even when R's language setting is not English.
- `xyTable()` now handles and reports `NA`s fixing PR#18654. Thanks to Heather Turner and Zhian Kamvar for report and patch.
- `as(*, "raw")` now works as documented, thanks to Mikael Jagan's PR#18795.
- Informational messages of e.g., `print(1:1e4, max=1000)`, now correctly mention `max` in addition to `getOption("max.print")`.
- `rowSums(A, dims = dd)`, `colMeans(..)`, etc, give a more helpful error message when `dd` is not of length one, thanks to Michael Chirico's PR#18811.
- `seq.Date()` no longer explicitly coerces results from integer to double, analogously with `seq.default()`, `seq.int()` and `seq.POSIXt()`, resolving a *modified* PR#18782.
- `axisTicks(usr, ...)` documentation clarification for `log = TRUE`, fixing bug PR#18821 thanks to Duncan Murdoch.
- `debug()` and `debugonce(fun)` now also accept a string `fun` when it names an S4 generic, fixing PR#18822 thanks to Mikael Jagan.
- `debugonce(<S4-simple-body>, signature=*)` now works correctly when "called twice", fixing PR#18824 thanks to Mikael Jagan.
- `format(dtime, digits=* / format=*)` is more consistent when the `POSIXt` date-time object `dtime` has fractional (non integer) seconds. Fixes PR#17350, thanks to new contributions by LatinR's 'R Dev Day' participants, Heather Turner and Dirk Eddelbuettel; also fixes more cases, notably when `format` contains '`%OS<nodigit>`'.
- `options(scipen = NULL)` and other invalid values now signal an error instead of invalidating ops relying on a finite integer value. Values outside the range -9 .. 9999 are now warned about and set to a boundary or to the default `0`, e.g., in case of an `NA`. This also fixes PR#16322.
- `cbind()` could segfault with `NULL` inputs. (Seen when R was built with `gcc-14`, LTO and C99 inlining semantics.)
- Fix segfault on `quartz()` from `grid.glyph()` call with `glyphInfo()` that describes non-existent font (PR#18784). Thanks to Tomek Gieorgijewski.
- `format()` etc, using `decimal.mark = s`, by default getting `s <- getOption("OutDec")`, signals a warning (to become an error in the future) when `s` is not a string with exactly one character.
- When `s <- getOption("OutDec")` is not a string of one character, a warning is signalled now whenever it is used in internal C code, notably when calling the default methods of `format()`.
- `pwilcox()` and `qwilcox()` now check for user interrupt less frequently.

- `summary(<stl>)` (which prints directly) finally gets the same `digits` default as the formatting printing of default `summary()` method results, and it is documented explicitly.
- `options(show.error.locations = TRUE)` once again shows the most recent known location when an error is reported. Setting its value to `"bottom"` is no longer supported. Numerical values are converted to logical.
- C API function `R_GetCurrentSrcref(skip)` now returns `srcref` entries correctly. (Note that there is also a change to the interpretation of `skip`; see the C-Level API entry above.)
- `tools::Rd2latex()` now removes leading and trailing spaces from `\alias` entries as documented, fixing indexing and linking hiccups in some PDF manuals.
- `R CMD Rd2pdf` can now render the package manual from a '`--latex`' installation also when the help contains figures.
- The argument of `as.environment()` is now named `x`, not `object`, as was always documented and shown when printing it; thanks to Gael Millot's PR#18849.
- When `R CMD check` aims at getting the time+date from a world clock, it is more robust against unexpected non-error results, thanks to Michael Chirico's PR#18852.
- The `tools::parseLatex()` parser made several parsing errors (PR#18855).
- Error messages produced by `tools::parseLatex()` are now more readable (PR#18855).
- `R CMD build <pkg>` excludes more file patterns when it tars the <pkg> directory, fixing both PR#18432 and PR#18434, for `vim` and `GNU Global emacs` users, thanks to Dirk Eddelbuettel's patch.
- `quantile()`'s default method gets an option `fuzz` to protect against floating point inaccuracy before rounding, thus fixing PR#15811 and, en passant, PR#17683.
- Printing arrays now also omits columns, rows and slices when desirable according to option `max.print`, or argument `max`, respectively, addressing most of the remaining part of PR#15027, thanks to Sherry Zhang's patch.
- `drop.terms(y ~ w, 1)` and similar now work, thanks to Benjamin Sommer's report in PR#18861 and collaboration with Heather Turner improving `reformulate()`.
- Many arguments which should be length-1 logical are checked more thoroughly. The most commonly seen errors are in `unlink(, recursive)`, `tempdir()` and the `na.rm` arguments of `max()`, `min()`, `sum()`, ...
  `grep()`, `strsplit()` and similar took non-`TRUE`/`FALSE` values of their logical arguments as `FALSE`, but these were almost always mismatches to unnamed arguments and are now reported as `NA`.
- `vignette("reshape")` is now also available on Windows.
- `trace(coerce, ..)` now works correctly, fixing PR#18823 thanks to Mikael Jagan.
- `R CMD check` now also reports bad symbols in package shared objects linked in from local static libraries (PR#18789).
- `factanal()` now works correctly also, e.g., for **GPArotation**, `oblimin()` rotations, fixing PR#18417, thanks to Coen Bernaards and others.
- Setting `attributes` on primitive functions is deprecated now and already an error in the development version of R. Changing the `environment` of a primitive does no longer happen and signals a warning.



Changes in R 4.4.3

Installation:

- R can be installed using C23 (for example with '`-std=gnu23`' or '`-std=gnu2x`') with recent compilers including `gcc` 12–14, Apple `clang` 15–16, LLVM `clang` 17–20 and Intel `icx` 2024.2.
  It can be installed with the upcoming (at the time of writing) `gcc` 15, which defaults to C23.

C-Level Facilities:

- The functions `R_strtod` and `R_atof` now allow hexadecimal constants without an exponent, for compatibility with their C99 versions (PR#18805).

Utilities:

- `R CMD build` and `R CMD check` now allow reference output for demo scripts ('`demo/`*demo*`.Rout.save`' files) to be shipped with the package, as proposed by Torsten Hothorn in PR#18816.

Bug Fixes:

- `kappa(A, exact=TRUE)` for singular `A` returns `Inf` more generally, fixing PR#18817 reported by Mikael Jagan.
- Fixed URLs of the sun spots (`sunspot.month` etc) data sets and mention future changes due to recalibration.
- The parser now accepts hexadecimal constants with a decimal point without an exponent (taken as `p0`) as documented in `?NumericConstants` (PR#18819).
- `rbind()` now works correctly when inputs include a raw vector and a logical, integer or double vector – previously the inclusion of the latter was garbled.
- `smooth.spline()` checks validity of its arguments `df.offset` and `penalty`: it could segfault if they were `NULL`.
- `isGeneric(<primitive>, fdef=*, getName=TRUE)` now also returns the name instead of just `TRUE`, fixing PR#18829 reported by Mikael Jagan.
- `isGeneric(fdef = print)` now works, fixing PR#18369 thanks to Mikael Jagan.
- `sort(x, method = "qsort")` made illegal accesses when `x` has length 0.
- `dir.create()` is protected against being passed an empty string as its `path` argument.
- Silent integer overflow could occur in the 'exact' computations for `fisher.test()` for unrealistic inputs: this is now an error.
- Some invalid C-level memory accesses are avoided for `loglin(, margin = NULL)`. `loglin(, param = TRUE)` no longer gives an error in corner cases such as a one-dimensional input.
- `dev.capabilities() $ events` now reports `"Idle"` if the device provides it, fixing PR#18836, thanks to Trevor Davis.
- `arima(.., seasonal = <wrong-vector>)` correctly errors now, ditto for `arima0()`, thanks to Norbert Kuder's report on the R-devel list.
- `binomial(<link>)$linkinv(eta)` and `.. $mu.eta(eta)` now also work for `"logit"` link when `is.integer(eta)`.
- `as.roman(x)` now should work platform independently, also for, e.g., `x = "IIIII"` (= V) and `x = "IIIIII"` (= VI).
- `R CMD Rd2pdf` works again on an installed package directory containing LaTeX help (from option '`--latex`'), thanks to a report by Peter Ruckdeschel.



Changes in R 4.4.2

C-Level Facilities:

- The S-compatibility macros `F77_COM` and `F77_COMDECL` defined in header '`R_ext/RS.h`' are deprecated and will be removed shortly. We could find no record of their use.

Bug Fixes:

- Mathlib function `lgammacor(x)` no longer warns about underflow to zero for large `x`.
- Text widths and heights were incorrectly reported by the Quartz device if the drawing context didn't exist yet (typically when drawing off-screen to a window that is yet to appear, see PR#18591).
- The Quartz device could segfault in cases where paths with spaces are used in the new glyph drawing API. Thanks to Tomek Gieorgijewski (PR#18758).
- On macOS in R CRAN builds, it is again possible to read little-endian UTF-16 text with a BOM from a connection using `encoding="UTF-16"`. Users building R from source should avoid using the system `libiconv` in macOS 14.1 and later.
- **methods**' internal `.requirePackage()` now re-enables primitive method dispatch when needed; thanks to Ivan Krylov for demystifying CRAN package check failures on the R-devel mailing list.



Changes in R 4.4.1

C-Level Facilities:

- Functions `R_atof` and `R_strtod` declared in header '`R_ext/Utils.h`' are now documented in 'Writing R Extensions' and so formally part of the API.
- The non-API entry points `Rf_setSVector`, `Rf_StringFalse`, `Rf_StringTrue` and `Rf_isBlankString` have been added to those reported by `R CMD check`.
- The new function `Rf_allocLang` is now available. This provides an alternative to the idiom of calling `Rf_allocList` followed by `SET_TYPEOF`.

Utilities:

- `R CMD check` now reports as warnings what `gfortran` calls 'Fortran 2018 deleted features', all of which have long been marked as 'obsolescent' and some of which were deleted in Fortran 2008 or earlier. Fortran compilers are no longer required to support these.

Bug Fixes:

- `as.numeric()`, `scan()`, `type.convert()` and other places which use the internal C function `R_strtod` now require a *non-empty* digit sequence in a decimal or binary exponent. This aligns with the C/POSIX standard for `strtod` and with `?NumericConstants`.
- `as.data.frame(m, make.names=NA)` now works correctly for a matrix `m` with `NA`'s in row names.
- The error message from `<POSIXlt>[["hour"]]` and similar now mentions `*[[, "hour"]]`, as wished for in PR#17409 and proposed by Michael Chirico.
- `qbinom()` and potentially `qpois()`, `qnbinom()`, no longer sometimes fail accurate inversion (of `pbinom()`, etc), thanks to Christopher Chang's report and patch in PR#18711.

- The internal help server on Windows can again serve requests sent in quick succession, fixing a regression in R 4.4.0.
- `debugcall(`*S3Generic*`())` now also works when a corresponding S4-generic version is in the **methods** cache (PR#18143).
- Package **tools**' `toTitleCase(ch0)` now returns `character(0)` when `ch0` is of zero length; fixing PR#18724, reported by David Hugh Jones.
- `R CMD check` is no longer broken (without a check result and no explanation in '`00check.log`') for a package which declares an invalid `VignetteBuilder` in '`DESCRIPTION`' but has no vignettes.



Changes in R 4.4.0

Significant User-Visible Changes:

- Startup banners, `R --version`, `sessionInfo()` and `R CMD check` no longer report `(64-bit)` as part of the platform as this is almost universal – the increasingly rare 32-bit platforms will still report `(32-bit)`.
  On Windows, ditto for window titles.
- `is.atomic(NULL)` now returns `FALSE`, as `NULL` is not an atomic vector. Strict back-compatibility would replace `is.atomic(foo)` by `(is.null(foo) || is.atomic(foo))` but should happen only sparingly.

New Features:

- The `confint()` methods for `"glm"` and `"nls"` objects have been copied to the **stats** package. Previously, they were stubs which called versions in package **MASS**. The **MASS** namespace is no longer loaded if you invoke (say) `confint(glmfit)`. Further, the `"glm"` method for `profile()` and the `plot()` and `pairs()` methods for class `"profile"` have been copied from **MASS** to **stats**. (`profile.nls()` and `plot.profile.nls()` were already in **stats**.)
- The `confint()` and `profile` methods for `"glm"` objects have gained a possibility to do profiling based on the Rao Score statistic in addition to the default Likelihood Ratio. This is controlled by a new `test =` argument.
- The `pairs()` method for `"profile"` objects has been extended with a `which =` argument to allow plotting only a subset of the parameters.
- The `"glm"` method for `anova()` computes test statistics and p-values by default, using a chi-squared test or an F test depending on whether the dispersion is fixed or free. Test statistics can be suppressed by giving argument `test` a false logical value.
- In `setRepositories()` the repositories can be set using their names via `name =` instead of index `ind =`.
- `methods()` and `.S3methods()` gain a `all.names` option for the (rare) case where functions starting with a '`.`' should be included.
- Serializations can now be interrupted (e.g., by *`Ctrl-C`* on a Unix-alike) if they take too long, e.g., from `save.image()`, thanks to suggestions by Ivan Krylov and others on R-devel.
- New startup option '`--max-connections`' to set the maximum number of simultaneous connections for the session. Defaults to 128 as before: allowed values up to 4096 (but resource limits may in practice restrict to smaller values).
- R on Windows (since Windows 10 2004) now uses the new Segment Heap allocator. This may improve performance of some memory-intensive applications.

- When R packages are built, typically by `R CMD build <pkg>`, the new `--user=<build_user>` option overrides the (internally determined) user name, currently `Sys.info()["user"]` or `LOGNAME`. This is a (modified) fulfillment of Will Landau's suggestion in PR#17530.
- `tools::testInstalledBasic()` gets new optional arguments `outDir` and `testSrcdir`, e.g., allowing to use it in a *builddir* `!=` *srcdir* setup, and in standard "binary" Windows installation **if** a source '`tests/`' folder is present.
- `range(<DT_with_Inf>, finite = TRUE)` now work for objects of class `"Date"`, `"POSIXct"`, and `"POSIXlt"` with infinite entries, analogously to `range.default()`, as proposed by Davis Vaughan on R-devel. Other `range()`-methods can make use of new function `.rangeNum()`.
- New `.internalGenerics` object complementing `.S3PrimitiveGenerics`, for documentation and low-level book-keeping.
- `grid()` now invisibly returns the x- and y- coordinates at which the grid-lines were drawn.
- `norm(., type)` now also works for complex matrices.
- `kappa(., exact = TRUE, norm = *)` now works for all norms and also for complex matrices. In symmetric / triangular cases, the new argument `uplo = "U" | "L"` allows the upper or lower triangular part to be specified.
- `memDecompress(type = "unknown")` recognizes compression in the default 'zlib' format as used by `memCompress(type = "gzip")`.
- `memCompress()` and `memDecompress()` will use the `libdeflate` library (https://github.com/ebiggers/libdeflate) if installed. This uses the same type of compression for `type = "gzip"` but is 1.5-2x faster than the system `libz` library on some common platforms: the speed-up may depend on the library version.
- `diff()` for objects of class `"Date"`, `"POSIXct"`, and `"POSIXlt"` accepts a `units` argument passed via `....`
- Dynamic help now does a much better job of rendering package '`DESCRIPTION`' metadata.
- `Rprof()` gains an `event` argument and support for elapsed (real) time profiling on Unix (PR#18076).
- `filled.contour()` gains a `key.border` argument.
- `tools::update_pkg_po()` gets arguments `pot_make` and `mo_make` for *not* re-making the corresponding files, and additionally a `verbose` argument.
- Hexadecimal string colour specifications are now accepted in short form, so, for example, we can use `"#123"`, which is equivalent to `"#112233"`.
  Thanks to MikeFC for the original idea and Ella Kaye, Malcolm Barrett, George Stagg, and Hanne Oberman for the patch.
- Plain-text help shows `\var` markup by angle brackets.
- The new experimental primitive function `declare()` is intended to eventually allow information about R code to be communicated to the interpreter, compiler, and code analysis tools. The syntax for declarations is still being developed.
- Functions `psmirnov()`, `qsmirnov()` and `rsmirnov()` in package **stats** have had argument `two.sided` renamed to `alternative`, to take into account that the permutation distributions of the one-sided statistics can be different in the case of ties. Consequence of PR#18582.
- `sort()` is now an implicit S4 generic in **methods**.
- Formatting and printing, `format(z)`, `print(z)`, of complex vectors `z` no longer zap relatively small real or imaginary parts to zero, fixing PR#16752. This is an API change, as it was documented previously to round real and imaginary parts together

on purpose, producing nicer looking output. As mentioned, e.g. in the PR, this change is compatible with many other "R-like" programming environments.
We have simplified the internal code and now basically format the real and imaginary parts independently of each other.

- New experimental functions `Tailcall()` and `Exec()` to support writing stack-space-efficient recursive functions.
- Where characters are attempted to be plotted by `pdf()`, `postscript()` and `xfig()` which are not in the selected 8-bit character set (most often Latin-1) and the R session is using a UTF-8 locale, the warning messages will show the UTF-8 character rather than its bytes and one dot will be substituted per character rather than per byte. (Platforms whose `iconv()` does transliteration silently plot the transliteration.)
  In a UTF-8 locale some transliterations are now done with a warning (e.g., dashes and Unicode minus to hyphen, ligatures are expanded, permille ('‰') is replaced by '`o/oo`'), although the OS may have got there first. These are warnings as they will continue to be replaced by dots in earlier versions of R.
- The matrix multiplication functions `crossprod()` and `tcrossprod()` are now also primitive and S3 generic, as `%*%` had become in R 4.3.0.
- `source()` and `example()` have a new optional argument `catch.aborts` which allows continued evaluation of the R code after an error.
- The non-Quartz `tiff()` devices allow additional types of compression if supported by the platform's '`libtiff`' library.
- The list of base and recommended package names is now provided by `tools::standard_package_names()`.
- `cairo_pdf()` and `cairo_ps()` default to `onefile = TRUE` to closer match `pdf()` and `postscript()`.
- New option `catch.script.errors` provides a documented way to catch errors and then continue in non-interactive use.
- `L %||% R` newly in **base** is an expressive idiom for the phrases `if(!is.null(L)) L else R` or `if(is.null(L)) R else L`.
- The return value from `warnings()` now always inherits from `"warnings"` as documented, now also in the case of no warnings where it previously returned `NULL`.
- `as.complex("1i")` now returns `0 + 1i` instead of `NA` with a warning.
- `z <- c(NA, 1i)` now keeps the imaginary part `Im(z[1]) == 0`, no longer coercing to `NA_complex_`. Similarly, `cumsum(z)` correctly sums real and imaginary parts separately, i.e., without "crosstalk" in case of `NA`s.
- On Alpine Linux `iconv()` now maps `"latin2"`, `"latin-2"`, `"latin9"` and `"latin-9"` to encoding names the OS knows about (case-insensitively).
- `iconv(sub = "Unicode")` now always zero-pads to four (hex) digits, rather than to 4 or 8. (This seems to have become the convention once Unicode restricted the number of Unicode points to $2^{21} - 1$ and so will never need more than 6 digits.)
- `NCOL(NULL)` now returns 0 instead of 1, for consistency with `cbind()`.
- The information for the Euro glyph missing from the Adobe '`.afm`' files for the Courier, Helvetica and Times families has been copied from their URW equivalents – this will improve vertical centring in the `pdf()` and `postscript()` devices.
- The included BLAS sources have been updated to those shipped with LAPACK version 3.12.0. The changes are almost entirely cosmetic.
- The included LAPACK sources have been updated to version 3.12.0 and some further double-complex routines added.
- There are new font families for the 2014–5 URW 2.0 fonts (see `?pdf`) which are included in recent versions of Ghostscript. These have font widths for most Greek

glyphs and a few others which were missing from the original versions (whose font families remain available for reproducibility, although Ghostscript-based viewers will render using the 2.0 versions).

- Improve the large-n efficiency of `as.matrix(<dist>)`, thanks an R contributors effort, notably by Tim Taylor and Heather Turner, see PR#18660.
- The default and `numeric` methods of `all.equal()` get a `check.class` option.
- `zapsmall()` gets new optional arguments, function `mFUN` and `min.d`, for extra flexibility; fulfills a wish in PR#18199. Also, it is now an implicit S4 generic in package **methods**.
- The Rd filter for `aspell()` gains an `ignore` argument.
- New generic function `sort_by()`, primarily useful for the `data.frame` method which can be used to sort rows of a data frame by one or more columns.
- The licence headers for the RPC code in '`src/extra/xdr`' have been updated to use the GPL-compatible licence published by Oracle America in 2010.
- New function `pkg2HTML()` in **tools** to create single-page HTML reference manuals for R packages.
- The byte code evaluator now uses less C stack space for recursive calls to byte-compiled functions. It also makes more of an effort to avoid allocations for scalar return values.
- New completion option `backtick` (disabled by default) allows non-syntactic completions to be wrapped in backquotes. This is currently only useful for Jupyter notebooks via the **IRkernel** package, and may cause problems for other backends.
- The numeric version creators now stop on invalid non-character version specifications.

Installation:

- The parser has been updated to work with `bison` 3.8.2, which is now used for the pre-generated parsers in '`gram.c`', `file.c`, and '`gramRd.c`'. A few parser error messages have changed, which may affect code that relies on exact messages.

Installation on a Unix-Alike:

- System valgrind headers are now required to use `configure` option '`--with-valgrind-instrumentation`' with value `1` or `2`.
- `configure` will warn if it encounters a 32-bit build, as that is nowadays almost untested.
- Environment variable `R_SYSTEM_ABI` is no longer used and so no longer recorded in '`etc/Renviron`' (it was not on Windows and was only ever used when preparing package **tools**).
- If the `libdeflate` library and headers are available, `libdeflate` rather than `libz` is used to (de)compress R objects in lazy-load databases. Typically tasks spend up to 5% of their time on such operations, although creating lazy-data databases is one of the exceptions.
  This can be suppressed if the library is available by the `configure` option '`--without-libdeflate-compression`'.
- `configure` option '`--enable-lto=check`' has not worked reliably since 2019 and has been removed.
- The minimum `autoconf` requirement for a maintainer build has been increased to `autoconf` 2.71.
  It is intended to increase this to 2.72 for R 4.5.0: the distributed '`configure`' file was generated using 2.72.

- The minimum version requirement for an external LAPACK has been reduced to 3.9.0.
- No default C++ compiler is set if no C++17 compiler is detected: there is no longer an automatic fallback to C++14 or C++11.
  Compilers from the last five years should have sufficient support: for others macros `CXX` and `CXXSTD` can be set in file '`config.site`' to provide a fallback if needed.
  The Objective-C++ compiler now by default uses the standard selected by R for C++ (currently C++17) rather than the default standard for the C++ compiler (which on macOS is still C++98).

Installation on macOS:

- A new `configure` option '`--with-newAccelerate`' makes use of Apple's 'new' BLAS / LAPACK interfaces in their Accelerate framework. Those interfaces are only available in macOS 13.3 or later, and building requires SDK 13.3 or later (from the Command Line Tools or Xcode 14.3 or later).
  By default the option uses new Accelerate for BLAS calls: to also use it for LAPACK use option '`--with-newAccelerate=lapack`'. The later interfaces provide LAPACK 3.9.1 rather than 3.2.1: 3.9.1 is from 2021-04 and does not include the improved algorithms introduced in LAPACK 3.10.0 (including for BLAS calls).

Installation on Windows:

- The makefiles and installer scripts for Windows have been tailored to Rtools44, an update of the Rtools43 toolchain. It is based on GCC 13 and newer versions of MinGW-W64, binutils and libraries (targeting 64-bit Intel CPUs). R-devel can no longer be built using Rtools43 without changes.
- Rtools44 has experimental support for 64-bit ARM (`aarch64`) CPUs *via* the LLVM 17 toolchain using `lld`, `clang`/`flang-new` and `libc++`.

Utilities:

- `R CMD check` notes when S4-style exports are used without declaring a strong dependence on package **methods**.
- `tools::checkRd()` (used by `R CMD check`) detects more problems with `\Sexpr`-based dynamic content, including bad nesting of `\Sexpr`s and invalid arguments.
- `tools::checkRd()` now reports Rd titles and section names ending in a period; this is ignored by `R CMD check` unless environment variable `_R_CHECK_RD_CHECKRD_MINLEVEL_` is set to -5 or smaller.
- `R CMD check` now notes Rd files without an `\alias`, as long documented in 'Writing R Extensions' §1.3.1. The check for a missing `\description` has been moved from `tools::checkRd()` to `tools::checkRdContents()`.
- `R CMD check` now visits '`inst/NEWS.Rd`' and OS-specific '`man`' subdirectories when checking Rd files.
- `tools::checkDocFiles()` and `tools::checkRdContents()` now also check internal Rd files by default, but "specially" (ignoring missing documentation of arguments).
- `R CMD Rdiff` gets option '`--useEx`'.
- `R CMD check` now warns on non-portable uses of Fortran `KIND` such as `INTEGER(KIND=4)` and `REAL(KIND=8)`.
  To see the failing lines set environment variable `_R_CHECK_FORTRAN_KIND_DETAILS_` to a true value.
- When checking Rd files, `R CMD check` now notes some of the "lost braces" that `tools::checkRd()` finds. Typical problems are Rd macros missing the initial backslash (e.g., '`code{...}`'), in-text set notation (e.g., '`{1, 2}`', where the braces

need escaping), and \itemize lists with \describe-style entries of the form \item{label}{description}.

- R CMD INSTALL (and hence check) will compile C++ code with -DR_NO_REMAP if environment variable _R_CXX_USE_NO_REMAP_ is set to a true value. It is planned that this will in future become the default for compiling C++.
- The new built-in Rd macro \dontdiff{} can be used to mark example code whose output should be ignored when comparing check output to reference output ('tests/Examples/*pkg*-Ex.Rout.save'). The \dontdiff tag, like \donttest, is *not* shown on the rendered help page, so provides a clean alternative to '## IGNORE_RDIFF_(BEGIN|END)' comments.
- R CMD build when there is no 'NAMESPACE', now uses the recommended exportPattern("^[^.]"), instead of exporting everything.
- R CMD check now warns about non-ASCII characters in the 'NAMESPACE' file (in addition to R files). Such packages are not portable and may fail to install on some platforms.

C-Level Facilities:

- Headers 'R_ext/Applic.h' and 'R-ext/Linpack.h' used to include 'R_ext/BLAS.h' although this was undocumented and unneeded by their documented entry points. They no longer do so.
- New function R_missing(), factored out from do_missing(), used to fix PR#18579.
- SEXP type S4SXP has been renamed to OBJSXP to support experimenting with alternative object systems. The S4SXP value can still be used in C code but is now deprecated. Based on contributions from the R Consortium's Object-Oriented Programming Working Group.
- New function pow1p(x,y) for accurate (1+x)^y.
- mkCharLenCE was incorrectly documented to take a size_t length but was implemented with int (and character strings in R are limited to $2^{31} - 1$ bytes).

Deprecated and Defunct:

- data() no longer handles zipped data from long-defunct (since R 2.13.0) '--use-zip-data' installations.
- The legacy graphics devices pictex() and xfig() are now deprecated. They do not support recent graphics enhancements and their font handling is rudimentary. The intention is to retain them for historical interest as long as they remain somewhat functional.
- Support for encoding = "MacRoman" has been removed from the pdf() and postscript() devices – this was a legacy encoding supporting classic macOS up to 2001 and no longer has universal libiconv support.
- is.R() is deprecated as no other S dialect is known to be in use (and this could only identify historical dialects, not future ones).
  Further information on calls can be obtained by setting the environment variable _R_DEPRECATED_IS_R_ to 'error' which turns the deprecation warning into an error and so by default gives a traceback. (This is done by R CMD check --as-cran.)
- UseMethod() no longer forwards local variables assigned in the generic function into method call environments before evaluating the method body. This makes method calls behave more like standard function calls and makes method code easier to analyze correctly.
- The twelve as.data.frame.<class>() methods which were deprecated only via _R_CHECK_AS_DATA_FRAME_EXPLICIT_METHOD_ and in R CMD check

`--as-cran` are formally deprecated now in favour of calling `as.data.frame()` or `as.data.frame.vector()`. The deprecation "check" now works also when `as.data.frame()` is S4 generic thanks to Ivan Krylov.

- The default method for the directional comparison operators `<`, `>`, `<=`, and `>=` now signals an error when one of the operands is a language object, i.e. a symbol or a call.
- For `terms.formula()`, deprecate `abb` and `neg.out` arguments *formally* in addition to just documenting it.

Bug Fixes:

- The **methods** package is more robust to not being attached to the search path. More work needs to be done.
- `pairwise.t.test()` misbehaved when subgroups had 0 DF for variance, even with `pool.sd = TRUE`. (PR#18594 by Jack Berry.)
- Probability distribution functions `[dpq]<distrib>(x, *)`, but also `bessel[IKJY](x, .)` now consistently preserve `attributes(x)` when `length(x) == 0`, e.g., for a 2×0 matrix, thanks to Karolis Koncevičius' report PR#18509.
- Group "Summary" computations such as `sum(1:3, 4, na.rm = 5, NA, 7, na.rm = LL)` now give an error instead of either `17` or `NN` for `LL` true or false, as proposed by Ivan Krylov on the R-devel mailing list. (This also means it is now an error to specify `na.rm` more than once.)
- `as.complex(x)` now returns `complex(real = x, imaginary = 0)` for *all* numerical and logical `x`, notably also for `NA` or `NA_integer_`.
- Directories are now omitted by `file.copy(, recursive = FALSE)` and in `file.append()` (PR#17337).
- `gsub()` and `sub()` are now more robust to integer overflow when reporting errors caused by too large input strings (PR#18346).
- Top-level handlers are now more robust to attempts to remove a handler whilst handlers are running (PR#18508).
- The handling of `Alt+F4` in dialogs created on Windows using GraphApp has been fixed (PR#13870).
- `density()` more consistently computes grid values for the FFT-based convolution, following Robert Schlicht's analysis and proposal in PR#18337, correcting density values typically by a factor of about 0.999. Argument `old.coords = TRUE` provides back compatibility.
- `palette.colors()` gains a `name` argument that defaults to `FALSE` controlling whether the vector of colours that is returned has names (where possible). PR#18529.
- `tools::xgettext()` no longer extracts the (non-translatable) class names from `warningCondition` and `errorCondition` calls.
- `S3method(<gen>, <class>, <func>)` in the '`NAMESPACE`' file now works (again) when `<func>` is visible from the namespace, e.g., imported, or in base.
- `getParseData(f)` now also works for a function defined in the first of several '`<pkg>/R/*.R`' source files, thanks to Kirill Müller's report and Duncan Murdoch's patch in PR#16756.
- Rd `\Sexpr` macros with nested `#ifdef` conditionals were not processed.
- A non-blocking connection with non-default encoding (such as a socket) now correctly returns from `readLines()` after new data has arrived also when its EOF had been reached previously. Thanks to Peter Meilstrup's report on R-devel and Ivan Krylov's report and patch proposal in PR#18555.

- `tools::checkRdContents()` failed to detect empty argument descriptions when they spanned multiple lines, including those generated by `prompt()`. These cases are now noted by `R CMD check`.
- Plain-text help no longer outputs spurious colons in the arguments list (for multi-line `\item` labels in the Rd source).
- `kappa()` and `rcond()` work correctly in more cases; `kappa(., norm = "2")` now warns that it computes the 1-norm with (default) `exact = FALSE`; prompted by Mikael Jagan's quite comprehensive PR#18543.
- Rd skeletons generated by `prompt()` or `promptData()` now use a dummy title (so `R CMD build` works). `tools::checkRdContents()` has been updated to detect such template leftovers, including from `promptPackage()`.
- When S4 method dispatch fails because no method was found, the error message now includes the signature argument names; thanks to Michael Chirico's proposal on the R-devel list.
- `withAutoprint({ .. })` now preserves `srcref`s previously lost, thanks to Andrew Simmons' report plus fix in PR#18572.
- `transform.data.frame()` no longer adjusts names; in particular, untransformed variables are kept as-is, including those with syntactically invalid names (PR#17890).
- The `keep.source` option for Rd `\Sexpr` blocks is no longer ignored.
- The `formula` methods for `t.test()` and `wilcox.test()` now catch when `paired` is passed, addressing PR#14359; use `Pair(x1, x2) ~ 1` for a paired test.
- The level reported in the `browser` prompt was often too large. It now shows the number of browser contexts on the stack.
- For `cbind()` and `rbind()`, the optional `deparse.level` argument is now properly passed to methods, thanks to Mikael Jagan's PR#18579 and comments there.
- Some error and warning messages for large ('long vector') `matrix(v, nr, nc)` and `dim(m) <- d` are now correct about sizes, using `long long` formatting, fixing PR#18612 (and more) reported by Mikael Jagan.
- `readChar(useBytes = TRUE)` now terminates strings even when the underlying connection uses extra spaces in the input buffer. This fixes problems with extra garbage seen with `gzip` connections, PR#18605.
- Named capture in PCRE regular expressions now works also with more than 127 named groups (PR#18588).
- Datetime functions are now robust against long jumps when dealing with internal time zone changes. This avoids confusing warnings about an invalid time zone, previously triggered by turning warnings into errors or handling them via `tryCatch` (PR#17966, PR#17780).
- Datetime functions now restore even an empty `TZ` environment variable after internal time zone changes (PR#17724). This makes results of datetime functions with this (typically unintentional) setting more predictable.
- `drop.terms(*)` now drops response as by default, `keep.response = FALSE`, fixing PR#18564 thanks to Mikael Jagan.
- `dummy.coef(.)` now also works for `lm()`-models with `character` categorical predictor variables rather than `factor` ones, fixing PR#18635 reported by Jinsong Zhao.
- `formals(f) <- formals(f)` now also works for a function w/o arguments and atomic *constant* `body(f)`.
- Correct `as.function(<invalid list>, .)`'s error message.
- `removeSource()` is yet more thorough in finding and removing `"srcref"` and the other source references from parsed R language chunks, fixing PR#18638 thanks to Andrew Simmons.

- `dgeom()` is more accurate now, notably when its result is very small, fixing PR#18642 thanks to the proposal of Morten Welinder, also improving other instances where C level `binom_raw(x, n, ..)` has `x == 0` or `x== n`.
- `warning()` with `options(warn = 1)` has improved output for multi-line messages.
- `axis.Date()` and `axis.POSIXct()` now respect the `par("lab")` setting for the number of `pretty()` intervals.
- Comparisons for language objects (which are based on deparsing) are now more careful about using accurate deparsed results (PR#18676).
- Plain-text help (`Rd2txt`) now correctly preserves blank lines following single-line `\dontrun` code.
- `<POSIXlt>[*]` no longer sets wrong `"balanced"` attribute, fixing PR#18681 thanks to Mikael Jagan.
- `str(<classed-call>)` now deparses the call as expected, fixing PR#18684, reported by Dave Slager.
- In Rd examples, code following the closing brace of a `\dontrun`, `\dontshow` or `\donttest` tag on the same line is no longer skipped when `R CMD check` runs the examples.
- `as.data.frame(matrix(*, ncol=0))` now gets valid `names()` and `colnames()`; reported by Davis Vaughan on the R-devel list.
- Internal Mathlib function `stirlerr(n)` is now almost fully (52-bit) accurate for all $n >= 5.9$ and more accurate also in the range $1 - -5.9$. This entails small ("after 12th decimal") changes in density functions, e.g., `dgamma()` in *some* regions of their support. The fix was partly prompted by Morten Welinder's PR#18640.
- Numbers like `9876543.2` are now considered non-integer by Mathlib internal `R_nonint()`, amending original fix of PR#15734.
- Rd comment lines no longer cause broken paragraphs in the rendered PDF and plain-text help. In code blocks, pure comment lines (starting with '`%`') no longer produce an empty line.
- `xtabs(Freq ~ .)` now consistently defaults to `na.action = na.pass`, using `na.rm = FALSE` (added as an argument) when summing over `Freq` (PR#17770).
- `tools::testInstalledPackage()` is no longer silent about failures from running examples or tests and its return code no longer ignores failures from checking vignettes.



Changes in R 4.3.3

New Features:

- `iconv()` now fixes up variant encoding names such as `"utf8"` case-insensitively.

Deprecated and Defunct:

- The legacy `encoding = "MacRoman"` is deprecated in `pdf()` and `postscript()`: support was incomplete in earlier versions of R.

Bug Fixes:

- Arguments are now properly forwarded to methods on S4 generics with ... in the middle of their formal arguments. This was broken for the case when a method introduced an argument but did not include ... in its own formals. Thanks to Hervé Pagès for the report PR#18538.

- Some invalid `file` arguments to `pictex()`, `postscript()` and `xfig()` opened a file called 'NA' rather than throw an error. These included `postscript(NULL)` (which some people expected to work like `pdf(NULL)`).
- Passing `filename = NA` to `svg()`, `cairo_pdf()`, `cairo_ps()` or the Cairo-based bitmap devices opened a file called 'NA': it now throws an error.
- `quartz(file = NA)` opened a file called 'NA', including when used as a Quartz-based bitmap device. It now gives an error.
- `rank(<long vector>)` now works, fixing PR#18617, thanks to Ilia Kats.
- `seq.int()` did not adequately check its `length.out` argument.
- `match(<POSIXct>, .)` is correct again for differing time zones, ditto for `"POSIXlt"`, fixing PR#18618 reported by Bastian Klein.
- `drop.terms(*, dropx = <0-length>)` now works, fixing PR#18563 as proposed by Mikael Jagan.
- `drop.terms(*)` keeps `+ offset(.)` terms when it should, PR#18565, and `drop.terms()` no longer makes up a response, PR#18566, fixing both bugs thanks to Mikael Jagan.
- `getS3method("t", "test")` no longer finds the `t.test()` function, fixing PR#18627.
- `pdf()` and `postscript()` support for the documented Adobe encodings `"Greek"` and `"Cyrilllic"` was missing (although the corresponding Windows' codepages could be used).
- Computations of glyph metric information for `pdf()` and `postscript()` did not take into account that transliteration could replace one character by two or more (only seen on macOS 14) and typically warned that the information was not known.
- `rank(x)` no longer overflows during integer addition, when computing rank average for largish but not-yet long vector `x`, fixing PR#18630, thanks to Ilia Kats.
- `list.files()` on Windows now returns also files with names longer that 260 bytes (the Windows limit is 260 characters). Previously, some file names particularly with 'East Asian' characters were omitted.
- `cov2cor(<0 x 0>)` now works, fixing PR#18423 thanks to Mikael Jagan and Elin Waring.
- `cov2cor(<negative diagonal>)` and similar now give one warning instead of two, with better wording, fixing PR#18424 thanks to Mikael Jagan.
- **tools**`::startDynamicHelp()` now ensures `port` is in proper range, fixing PR#18645.
- `pbeta(x, a,b)` is correct now for `x`=0 or 1 in the boundary cases where `a` or `b` or both are 0, fixing PR#18672 thanks to Michael Fay.
- `pmatch(x, table)` for large `table`, also called for data frame row selection, `dfrm[nm, ]`, is now interruptible, fixing PR#18656.
- `predict(<rank-deficient lm>, newdata=*)` fix computing of `nbasis`, see Russ Lenth's comment 29 in PR#16158.
- Added a work-around for a bug in macOS 14.3.1 and higher which prevents R plots in the Quartz Cocoa device from updating on screen.



Changes in R 4.3.2

New Features:

- The default initialization of the `"repos"` option from the '`repositories`' file at startup can be skipped by setting environment variable `R_REPOSITORIES` to `NULL` such that `getOption("repos")` is empty if not set elsewhere.

- `qr.X()` is now an implicit S4 generic in **methods**.
- `iconv(to = "ASCII//TRANSLIT")` is emulated using substitution on platforms which do not support it (notably Alpine Linux). This should give a human-readable conversion in ASCII on all platforms (rather than `NA_character_`).
- `trans3d()` gains options `continuous` and `verbose` addressing the problem of possible "wrap around" when projecting too long curves, as reported by Achim Zeileis in PR#18537.
- `tools::showNonASCII()` has been rewritten to work better on macOS 14 (which has a changed implementation of `iconv()`).
- `tiff(type = "quartz")` (the default on macOS) now warns if `compression` is specified: it continues to be ignored.

Installation on a Unix-Alike:

- There is some support for building with Intel's LLVM-based compilers on '`x86_64`' Linux, such as (C) `icx`, (C++) `ipcx` and (Fortran) `ifx` from oneAPI 2023.x.y.
- There is support for using LLVM's `flang-new` as the Fortran compiler from LLVM 16.0.x (preferably 17.0.0 or later).

Utilities:

- `R CMD check` reports the use of the Fortran 90 random number generator `RANDOM_NUMBER()` and the subroutines to initialize it.
  'Writing R Extensions' has example code to use R's RNGs from Fortran.

Bug Fixes:

- `substr(x, n, L) <- cc` now works (more) correctly for multibyte UTF-8 strings `x` when `L > nchar(x)`, thanks to a report and patch by 'Architect 95'.
- `contrib.url(character())` now returns 0-length `character()` as documented, which also avoids spurious warnings from `available.packages()` et al. in the edge case of an empty vector of repository URLs.
- `readChar(., 4e8)` no longer fails, thanks to Kodi Arfer's report (PR#18557).
- `lapply(<list>, as.data.frame)` no longer warns falsely for some base vector components.
- Communication between parent and child processes in the `multicore` part of `parallel` could fail on platforms that do not support an arbitrarily large payload in system functions `read()`/`write()` on pipes (seen on macOS where a restriction to `INT_MAX` bytes is documented, without doing a partial read unlike Linux). The payload is now split into 1Gb chunks to avoid that problem. (PR#18571)
- `qqplot(x,y, conf.level=.)` gives better confidence bounds when `length(x) != length(y)`, thanks to Alexander Ploner's report and patch proposal (PR#18557).
- `norm(<0-length>, "2")` now gives zero instead of an error, as all the other norm types, thanks to Mikael Jagan's PR#18542.
- Build-stage Rd macros `\packageAuthor` and `\packageMaintainer` now process '`Authors@R`', fixing '`NA`' results when the package '`DESCRIPTION`' omits '`Author`' and '`Maintainer`' fields.
- Formatting and printing complex numbers could give things like '`0.1683-0i`' because of rounding error: '`-0i`' is now replaced by '`+0i`'.
- `postscript()` refused to accept a `title` comment containing the letter "W" (PR#18599).
- `isoreg(c(1,Inf))` signals an error instead of segfaulting, fixing PR#18603.
- `tiff(type = "Xlib")` was only outputting the last page of multi-page plots.

- `tools::latexToUtf8()` again knows about '`\~{n}`' and other letters with tilde, fixing a regression in R 4.3.0, and about '`\^{i}`' as an alternative to '`\^{\i}`' (similarly with other accents). Furthermore, LaTeX codes for accented I letters are now correctly converted, also fixing related mistakes in `tools::encoded_text_to_latex()`.
- `tar(*, tar = "internal")` no longer creates out-of-spec tar files in the very rare case of user or group names longer than 32 bytes, fixing PR#17871 with thanks to Ivan Krylov.
- When using the "internal" timezone datetime code, adding a fraction of a second no longer adds one second, fixing PR#16856 from a patch by Ivan Krylov.
- `tools::checkRd()` no longer produces spurious notes about "unnecessary braces" from multi-line Rd results of `\Sexpr` macros.



Changes in R 4.3.1

C-Level Facilities:

- The C-level API version of R's `integrate()`, `Rdqags()` in '`Applic.h`', now returns the correct number of integrand evaluations `neval`, fixing PR#18515 reported and diagnosed by Stephen Wade.
- The C prototypes for LAPACK calls `dspgv` and `dtptrs` in '`R_ext/Lapack.h`' had one too many and one too few character length arguments — but this has not caused any known issues. To get the corrected prototypes, include

```
#include <Rconfig.h> // for PR18534fixed
#ifdef PR18534fixed
# define usePR18534fix 1
       #endif
#include <R_ext/Lapack.h>
```

  in your C/C++ code (PR#18534).

Installation:

- Many of the checks of esoteric Internet operations and those using unreliable external sites have been moved to a new target that is not run by default and primarily intended for the core developers. To run them use

```
cd tests; make test-Internet-dev
```

Bug Fixes:

- `.S3methods()`, typically called from `methods()`, again marks methods from package **base** as `visible`.
  Also, the visibility of non-**base** methods is again determined by the method's presence in `search()`.
- `tools::Rdiff()` is now more robust against invalid strings, fixing installation tests on Windows without Rtools installed (PR#18530).
- Fix (new) bug in `hcl.colors(2, *)`, by Achim Zeileis (PR#18523).
- `head(., <illegal>)` and `tail(..)` now produce more useful `"Error in ...."` error messages, fixing PR#18362.
- Package code syntax on Windows is checked in UTF-8 when UTF-8 is the native encoding.

- `na.contiguous(x)` now also returns the first run, when it is at the beginning and there is a later one of the same length; reported to R-devel, including a fix, by Georgi Boshnakov. Further, by default, it modifies only an existing `attr(*,"tsp")` but otherwise no longer sets one.
- `chol(<not pos.def>, pivot = <T|F>)` now gives a correct error or warning message (depending on `pivot`), thanks to Mikael Jagan's (PR#18541).



Changes in R 4.3.0

Significant User-Visible Changes:

- Calling `&&` or `||` with LHS or (if evaluated) RHS of length greater than one is now always an error, with a report of the form

  ```
  'length = 4' in coercion to 'logical(1)'
  ```

  Environment variable `_R_CHECK_LENGTH_1_LOGIC2_` no longer has any effect.

New Features:

- The included BLAS sources have been updated to those shipped with LAPACK version 3.10.1. (This caused some platform-dependent changes to package check output.) And then to the sources from LAPACK version 3.11.0 (with changes only to double complex subroutines).
- The included LAPACK sources have been updated to include the four Fortran 90 routines rather than their Fortran 77 predecessors. This may give some different signs in SVDs or eigendecompositions.. (This completes the transition to LAPACK 3.10.x begun in R 4.2.0.)
- The LAPACK sources have been updated to version 3.11.0. (No new subroutines have been added, so this almost entirely bug fixes: Those fixes do affect some computations with `NaN`s, including R's `NA`.)
- The parser now signals *classed* errors, notably in case of the pipe operator `|>`. The error object and message now give line and column numbers, mostly as proposed and provided by Duncan Murdoch in PR#18328.
- `toeplitz()` is now generalized for asymmetric cases, with a `toeplitz2()` variant.
- `xy.coords()` and `xyz.coords()` and consequently, e.g., `plot(x,y, log = "y")` now signal a *classed* warning about negative values of y (where `log(.)` is `NA`). Such a warning can be specifically suppressed or caught otherwise.
- Regular expression functions now check more thoroughly whether their inputs are valid strings (in their encoding, e.g. in UTF-8).
- The performance of `grep()`, `sub()`, `gsub()` and `strsplit()` has been improved, particularly with `perl = TRUE` and `fixed = TRUE`. Use of `useBytes = TRUE` for performance reasons should no longer be needed and is discouraged: it may lead to incorrect results.
- `apropos()` gains an argument `dot_internals` which is used by the completion (`help(rcompgen)`) engine to also see **base** internals such as `.POSIXct()`.
- Support in `tools::Rdiff()` for comparing uncompressed PDF files is further reduced – see its help page.
- `qqplot(x, y, ...)` gains `conf.level` and `conf.args` arguments for computing and plotting a confidence band for the treatment function transforming the distribution of `x` into the distribution of `y` (Switzer, 1976, *Biometrika*). Contributed by Torsten Hothorn.

- Performance of `package_dependencies()` has been improved for cases when the number of dependencies is large.
- Strings newly created by `gsub()`, `sub()` and `strsplit()`, when any of the inputs is marked as `"bytes"`, are also marked as `"bytes"`. This reduces the risk of creating invalid strings and accidental substitution of bytes deemed invalid.
- Support for `readLines(encoding = "bytes")` has been added to allow processing special text files byte-by-byte, without creating invalid strings.
- `iconv(from = "")` now takes into account any declared encoding of the input elements and uses it in preference to the native encoding. This reduces the risk of accidental creation of invalid strings, particularly when different elements of the input have different encoding (including `"bytes"`).
- Package repositories in `getOption("repos")` are now initialized from the '`repositories`' file when **utils** is loaded (if not already set, e.g., in '`.Rprofile`'). (From a report and patch proposal by Gabriel Becker in PR#18405.)
- `compactPDF()` gets a `verbose` option.
- `type.convert()` and hence `read.table()` get new option `tryLogical = TRUE` with back compatible default. When set to false, converts `"F"` or `"T"` columns to character.
- Added new unit prefixes `"R"` and `"Q"` for abbreviating (unrealistically large) sizes beyond 10^{27} in `standard = "SI"`, thanks to Henrik Bengtsson's PR#18435.
- `as.data.frame()`'s default method now also works fine with atomic objects inheriting from classes such as `"roman"`, `"octmode"` and `"hexmode"`, thus fulfilling the wish of PR#18421, by Benjamin Feakins.
- The `as.data.frame.vector()` utility now errors for wrong-length `row.names`. It warned for almost six years, with "Will be an error!".
- `sessionInfo()` now also contains `La_version()` and reports codepage and timezone when relevant, in both `print()` and `toLatex()` methods which also get new option `tzone` for displaying timezone information when `locale = FALSE`.
- New function `R_compiled_by()` reports the C and Fortran compilers used to build R, if known.
- `predict(<lm>, newdata = *)` no longer unnecessarily creates an `offset` of all `0`s.
- `solve()` for complex inputs now uses argument `tol` and by default checks for 'computational singularity' (as it long has done for numeric inputs).
- `predict(<rank-deficient lm>, newdata=*)` now obeys a new argument `rankdeficient`, with new default `"warnif"`, warning only if there are non-estimable cases in `newdata`. Other options include `rankdeficient = "NA"`, predicting `NA` for non-estimable `newdata` cases. This addresses PR#15072 by Russ Lenth and is based on his original proposal and discussions in PR#16158 also by David Firth and Elin Waring. Still somewhat experimental.
- `Rgui` console implementation now works better with the `NVDA` screen reader when the full blinking cursor is selected. The underlying improvements in cursor handling may help also other screen readers on Windows.
- The drop-field control in GraphApp can now be left with the TAB key and all controls can be navigated in the reverse order using the Shift+TAB key, improving accessibility of the `Rgui` configuration editor.
- `qnorm(<very large negative>, log.p=TRUE)` is now fully accurate (instead of to "only" minimally five digits).
- `demo(error.catching)` now also shows off `withWarnings()` and `tryCatchWEMs()`.
- As an experimental feature the placeholder `_` can now also be used in the `rhs` of a forward pipe `|>` expression as the first argument in an extraction call, such as `_$coef`. More generally, it can be used as the head of a chain of extractions, such as `_$coef[[2]]`.

- Spaces in the environment variable used to choose the R session's temporary directory (`TMPDIR`, `TMP` and `TEMP` are tried in turn) are now fatal. (On Windows the 'short path' version of the path is tried and used if that does not contain a space.)
- `all.equal.numeric()` gets a new optional switch `giveErr` to return the numeric error as attribute. Relatedly, `stopifnot(all.equal<some>(a, b, ..))` is as "smart" now, as `stopifnot(all.equal(....))` has been already, thus allowing customized `all.equal<Some>()` wrappers.
- R on Windows is now able to work with path names longer than 260 characters when these are enabled in the system (requires at least Windows 10 version 1607). Packages should be updated to work with long paths as well, instead of assuming `PATH_MAX` to be the maximum length. Custom front-ends and applications embedding R need to update their manifests if they wish to allow this feature. See https://blog.r-project.org/2023/03/07/path-length-limit-on-windows for more information.
- 'Object not found' and 'Missing argument' errors now give a more accurate error context. Patch provided by Lionel Henry in PR#18241.
- The `@` operator is now an S3 generic. Based on contributions by Tomasz Kalinowski in PR#18482.
- New generic `chooseOpsMethod()` provides a mechanism for objects to resolve cases where two suitable methods are found for an Ops Group Generic. This supports experimenting with alternative object systems. Based on contributions by Tomasz Kalinowski in PR#18484.
- `inherits(x, what)` now accepts values other than a simple character vector for argument `what`. A new generic, `nameOfClass()`, is called to resolve the class name from `what`. This supports experimenting with alternative object systems. Based on contributions by Tomasz Kalinowski in PR#18485.
- Detection of BLAS/LAPACK in use (`sessionInfo()`) with FlexiBLAS now reports the current backend.
- The `"data.frame"` method for `subset()` now warns about extraneous arguments, typically catching the use of '`=`' instead of '`==`' in the `subset` expression.
- Calling `a:b` when numeric `a` or `b` is longer than one may now be made into an error by setting environment variable `_R_CHECK_LENGTH_COLON_` to a true value, along the proposal in PR#18419 by Henrik Bengtsson.
- `density(x, weights = *)` now warns if automatic bandwidth selection happens without using `weights`; new optional `warnWbw` may suppress the warning. Prompted by Christoph Dalitz' PR#18490 and its discussants.
- `rm(list = *)` is faster and more readable thanks to Kevin Ushey's PR#18492.
- The `plot.lm()` function no longer produces a normal Q-Q plot for GLMs. Instead it plots a half-normal Q-Q plot of the absolute value of the standardized deviance residuals.
- The `print()` method for class `"summary.glm"` no longer shows summary statistics for the deviance residuals by default. Its optional argument `show.residuals` can be used to show them if required.
- The `tapply()` function now accepts a data frame as its `X` argument, and allows `INDEX` to be a formula in that case. `by.data.frame()` similarly allows `INDICES` to be a formula.
- The performance of `df[j] <- value` (including for missing `j`) and `write.table(df)` has been improved for data frames `df` with a large number of columns. (Thanks to Gabriel Becker's PR#18500, PR#18503 and discussants, prompted by a report from Toby Dylan Hocking on the R-devel mailing list.)

- The matrix multiply operator `%*%` is now an S3 generic, belonging to new group generic `matrixOps`. From Tomasz Kalinowski's contribution in PR#18483.
- New function `array2DF()` to convert arrays to data frames, particularly useful for the list arrays created by `tapply()`.

Dates and Times:

- On platforms where (non-UTC) datetimes before 1902 (or before 1900 as with system functions on recent macOS) are guessed by extrapolating time zones from 1902-2037, there is a warning at the first use of extrapolation in a session. (As all time zones post 2037 are extrapolation, we do not warn on those.)
- (Platforms using '`--with-internal-tzone`', including Windows and by default macOS). How years are printed in dates or date-times can be controlled by environment variable `R_PAD_YEARS_BY_ZERO`. The default remains to pad to 4 digits by zeroes, but setting value '`no`' gives no padding (as used by default by `glibc`).
- `strftime()` tries harder to determine the offset for the `"%z"` format, and succeeds on the mainstream R platforms.
- `strftime()` has a limit of 2048 bytes on the string produced – attempting to exceed this is an error. (Previously it silently truncated at 255 bytes.)
- `sessionInfo()` records (and by default prints) the system time zone as part of the locale information. Also, the source (system/internal) of the date-time conversion and printing functions.
- Objects of class `"POSIXlt"` created in this version of R always have 11 components: component `zone` is always set, and component `gmtoff` is set for times in UTC and usually set on the (almost all) platforms which have C-level support, otherwise is `NA`.
- There are comprehensive validity checks on the structure of objects of class `"POSIXlt"` when converting (including formatting and printing). (This avoids mis-conversions of hand-crafted objects.)
- There is some support for using the native date-time routines on macOS: this is only viable on recent versions (e.g. 12.6 and 13) and does get wrong some historical changes (before 1900, during WWII). Use of '`--with-internal-tzone`' remains the default.
- `as.POSIXct(<numeric>)` and `as.POSIXlt(.)` (without specifying `origin`) now work. So does `as.Date(<numeric>)`.
- `as.Date.POSIXct(., tz)` now treats several `tz` values, notably `"GMT"` as equivalent to `"UTC"`, proposed and improved by Michael Chirico and Joshua Ulrich in PR#17674.
- Experimental `balancePOSIXlt()` utility allows using "ragged" and or out-of-range `"POSIXlt"` objects more correctly, e.g., in subsetting and subassignments. Such objects are now documented. Complemented by the low-level `unCfillPOSIXlt()` utility. More experimentally, a `"POSIXlt"` object may have an attribute `"balanced"` indicating if it is known to be filled or fully balanced.
- Functions `axis.Date()` and `axis.POSIXct()` are rewritten to gain better default tick locations and better default formats via the corresponding `pretty()` methods. Thanks to Swetlana Herbrandt.
- The mapping of Windows' names for time zones to IANA's 'Olson' names has been updated. When ICU is available (it is by default), it is used to get a mapping for the current region set in Windows. This can be overridden by setting environment variable `TZ` to the desired Olson name — see `OlsonNames()` for those currently available.

Graphics:

- The graphics engine version, `R_GE_version`, has been bumped to `16` and so packages that provide graphics devices should be reinstalled.

- The **grDevices** and **grid** packages have new functions for rendering typeset glyphs, primarily: `grDevices::glyphInfo()` and `grid::grid.glyph()`.
  Rendering of typeset glyphs is only supported so far on the Cairo-based graphics devices and on the `pdf()` and `quartz()` devices.
- The defined behaviour for `"clear"` and `"source"` compositing operators (via `grid::grid.group()`) has been changed (to align better with simple interpretation of original Porter-Duff definitions).
- Support for gradients, patterns, clipping paths, masks, groups, compositing operators, and affine transformations has been added to the `quartz()` device.

Installation on a Unix-Alike:

- A system installation of generic LAPACK 3.10.0 or later will be preferred to the version in the R sources.
  `configure` option '`--with-lapack=no`' (equivalently '`--without-lapack`') forces compilation of the internal LAPACK sources.
  If '`--with-lapack`' is not specified, a system `liblapack` is looked for and used if it reports version 3.10.0 or later and does not contain BLAS routines.
  Packages using LAPACK will need to be reinstalled if this changes to using an external library.
- On '`aarch64`' Linux platforms using GCC, `configure` now defaults to '`-fPIC`' (instead of '`-fpic`'), as desired in PR#18326.
- `configure` now checks conversion of datetimes between `POSIXlt` and `POSIXct` around year 2020. Failure (which has been seen on platforms missing `tzdata`) is fatal.
- If `configure` option '`--with-valgrind-instrumentation`' is given value `1` or `2`, option '`--with-system-valgrind-headers`' is now the default and ignored (with a warning). It is highly recommended that the system headers are installed alongside `valgrind`: they are part of its packaging on some Linux distributions and packaged separately (e.g. in the '`valgrind-devel`' RPM) on others. `configure` will give a warning if they are not found.
  The system headers will be required in a future release of R to build with `valgrind` instrumentation.
- `libcurl` 8.x is now accepted by `configure`: despite a change in major version number it changes neither API nor ABI.

Installation on Windows:

- The makefiles and installer scripts for Windows have been tailored to Rtools43, an update of the Rtools42 toolchain. It is based on GCC 12 and newer versions of MinGW-W64, binutils and libraries. At this time R-devel can still be built using Rtools42 without changes, but when R-devel is installed via the installer, it will by default look for Rtools43.
- Old make targets `rsync-extsoft` and 32-bit ones that are no longer needed have been removed.
- Default builds (including for packages) no longer select C99. Thus the C standard used is the default for the compiler, which for the toolchain in '`Rtools43`' is C17. (This is consistent with Unix builds.)

Package Installation:

- The default C++ standard has been changed to C++17 where available (which it is on all currently checked platforms): if not C++14 or C++11 is used if available otherwise C++ is not supported.

- `USE_FC_LEN_T` is the default: this uses the correct (compiler-dependent) prototypes for Fortran BLAS/LAPACK routines called from C/C++, and requires adjustment of many such calls – see 'Writing R Extensions' §6.6.1.
- There is initial support for C++23 as several compilers are now supporting `-std=c++23` or `-std=c++2b` or similar. As for C++20, there no additional `configure` checks for C++23 features beyond a check that the compiler reports a `__cplusplus` value greater than that in the C++20 standard. C++ feature tests should be used.
- There is support for a package to indicate the version of the C standard which should be used to compile it, and for the installing user to specify this. In most cases R defaults to the C compiler's default standard which is C17 (a 'bug-fix' of C11) – earlier versions of R or compilers may have defaulted to C99.
  Current options are:
  USE_C17 Use a standard that is at most C17. The intention is to allow legacy packages to still be installed when later C standards become the default, including packages using new keywords as identifiers or with K&R-style function declarations. This will use C17 if available, falling back to C11.
  USE_C90 Use the C90 (aka C89) standard. (As that standard did not require compilers to identify that version, all we can verify is that the compiler does not claim to be using a later standard. It may accept C99 features – for example `clang` accepts // to make comments.)
  USE_C99 Use the C99 standard. This should be rarely needed – it avoids the few new features of C11/C17 which can be useful if a package assumes them if C17 is specified and they are not implemented.
  USE_C23 Use C23 (or in future, later). Compiler/library support for C23 is still being implemented, but LLVM clang from 15.0.0 and GCC from 13 have quite extensive support.
  These can be specified as part of the '`SystemRequirements`' field in the package's '`DESCRIPTION`' file or *via* options '`--use-C17`' and so on of `R CMD INSTALL` and `R CMD SHLIB`.
  For further details see "Writing R Extensions" §1.2.5.
- (Windows) A '`src/Makefile.ucrt`' or '`src/Makefile.win`' file is now included after '*R_HOME*`/etc`*R_ARCH*`/Makeconf`' and so no longer needs to include that file itself. Installation of a package with such a file now uses a site '`Makevars`' file in the same way as a package with a '`src/Makevars.win`' file would.
- `configure` is now passed crucial variables such as `CC` and `CFLAGS` in its environment, as many packages were not setting them (as documented in 'Writing R Extensions' §1.2).
  This has most effect where `configure` is used to compile parts of the package – most often by `cmake` or `libtool` which obfuscate the actual compile commands used.
  Also used for `configure.win` and `configure.ucrt` on Windows.

FORTRAN Flags:

- The flag '`-fno-optimize-sibling-calls`' is no longer forced for `gfortran` 7 and later. It should no longer be needed now using 'hidden' character-length arguments when calling BLAS/LAPACK routines from C/C++ is the default even for packages. (Unless perhaps packages call Fortran code from C/C++ without using R's headers and without allowing for these arguments.)

C-Level Facilities:

- The deprecated S-compatibility macros `DOUBLE_*` in '`R_ext/Constants.h`' (included by '`R.h`') have been removed.

- The deprecated legacy typedefs of `Sint` and `Sfloat` in header '`R.h`' are no longer defined, and that header no longer includes header '`limits.h`' from C nor '`climits`' from C++.
- New macro `CAD5R()` is provided in '`Rinternals.h`' and used in a few places in the R sources.
- ALTREP now supports `VECSXP` vectors. Contributed by Gabor Csardi in PR#17620.
- The `Rcomplex` definition (in header '`R_ext/Complex.h`') has been extended to prevent possible mis-compilation when interfacing with Fortran (PR#18430). The new definition causes compiler warnings with static initializers such as `{1, 2}`, which can be changed to `{.r=1,.i=2}`.
  Using the new definition from C++ depends on compiler extensions supporting C features that have not been incorporated into the C++ standards but are available in `g++` and `clang++`: this may result in C++ compiler warnings but these have been worked around for recent versions of common compilers (GCC, Apple/LLVM clang, Intel).
  It is intended to change the inclusion of header '`R_ext/Complex.h`' by other R headers, so C/C++ code files which make use of `Rcomplex` should include that header explicitly.

Utilities:

- `R CMD check` does more checking of package '`.Rd`' files, warning about invalid email addresses and (some) invalid URIs and noting empty '`\item`' labels in description lists.
- `R CMD check` now also reports problems when reading package news in md (file '`NEWS.md`') and (optionally) plain text (file '`NEWS`') formats.
- `_R_CHECK_TIMINGS_` defaults to a value from the environment even for `R CMD check --as-cran`; this allows for exceptionally fast or slow platforms.
  It now applies to checking PDF and HTML versions of the manuals, and 'checking CRAN incoming feasibility'.
- `R CMD check` can optionally (but included in '`--as-cran`') check whether HTML math rendering *via* KaTeX works for the package '`.Rd`' files.
- Non-interactive debugger invocations can be trapped by setting the environment variable `_R_CHECK_BROWSER_NONINTERACTIVE_` to a true value. This is enabled by `R CMD check --as-cran` to detect the use of leftover `browser()` statements in the package.
- The use of `sprintf` and `vsprintf` from C/C++ has been deprecated in macOS 13 and is a known security risk. `R CMD check` now reports (on all platforms) if their use is found in compiled code: replace by `snprintf` or `vsnprintf` respectively. [**NB:** whether such calls get compiled into the package is platform-dependent.]
- Where recorded at installation, `R CMD check` reports the C and Fortran compilers used to build R.
  It reports the OS in use (if known, as given by `osVersion`) as well as that R was built for.
  It notes if a C++ standard was specified which is older than the current default: many packages have used C++11 to mean 'not C++98' — as C++11 is the minimum supported since R 4.0.0, that specification can probably be removed.
- `R CMD INSTALL` reports the compilers (and on macOS, the SDK) used, and this is copied to the output of `R CMD check`.
  Where a C++ standard is specified, it is reported.
- `R CMD check`'s 'checking compilation flags in Makevars' has been relaxed to accept the use of flags such as '`-std=f2008`' in '`PKG_FFLAGS`'.

- `tools::buildVignettes()` has a new argument `skip`, which is used by `R CMD check` to skip (and note) vignettes with unavailable '`\VignetteDepends`' (PR#18318).
- New generic `.AtNames()` added to enable class-specific completions after `@`. The formerly internal function `findMatches()` is now exported, mainly for use in methods for `.DollarNames()` and `.AtNames()`.

Deprecated and Defunct:

- `default.stringsAsFactors()` is defunct.
- Calling `as.data.frame.<class>()` directly (for 12 atomic classes) is going to be formally deprecated, currently activated by setting the environment variable `_R_CHECK_AS_DATA_FRAME_EXPLICIT_METHOD_` to non-empty, which also happens in `R CMD check --as-cran`.

Bug Fixes:

- Hashed `environment`s with sizes less than 5 can now grow. (Reported to R-devel by Duncan Garmonsway.)
- `as.character(<Rd>, deparse = TRUE)` failed to re-escape curly braces in LaTeX-like text. (Reported by Hadley Wickham in PR#18324.)
- `library()` now passes its `lib.loc` argument when requiring `Depends` packages; reported (with fix) in PR#18331 by Mikael Jagan.
- `R CMD Stangle`: improved message about 'Output' files.
- `head(x, n)` and `tail(x, n)` now signal an error if `n` is not numeric, instead of incidentally "working" sometimes returning all of `x`. Reported and discussed by Colin Fay, in PR#18357.
- The `"lm"` method for `summary()` now gives the correct F-statistic when the model contains an offset. Reported in PR#18008.
- `C()` and `` `contrasts<-`() `` now preserve factor level names when given a function object (as opposed a function name which did preserve names). Reported in PR#17616.
- `c(a = 1, 2)[[]]` no longer matches `2` but rather signals a *classed* error. Reported and analysed by Davis Vaughan in PR#18367, a duplicate of PR#18004, by Jan Meis et al. For consistency, `NULL[[]]` is also erroneous now. `x[[]] <- v` gives an error of the same class `"MissingSubscriptError"`.
- The `relist()` function of **utils** now supports `NULL` elements in the skeleton (PR#15854).
- `ordered(levels = *)` (missing `x`) now works analogously to `factor(, ordered=TRUE)`; reported (with fix) by Achim Zeileis in PR#18389.
- User-defined Rd macro definitions can now span multiple lines, thanks to a patch from Duncan Murdoch. Previously, the Rd parser silently ignored everything after the first line.
- Plain-text help (`tools::Rd2txt()`) now preserves an initial blank line for text following description list items.
- `tools::Rd2HTML()` and `tools::Rd2latex()` no longer split `\arguments` and `\value` lists at Rd comments.
- `tools::Rd2latex()` now correctly handles optional text outside `\item`s of argument lists as well as bracketed text at the beginning of sections, e.g., `\value{[NULL]}`.
- `as.character(<POSIXt>)` now behaves more in line with the methods for atomic vectors such as numbers, and is no longer influenced by `options()`. Ditto for `as.character(<Date>)`. The `as.character()` method gets arguments `digits` and `OutDec` with defaults *not* depending on `options()`. Use of `as.character(*, format = .)` now warns.

- Similarly, the `as.character.hexmode()` and `*.octmode()` methods also behave as `good citizen` methods and back compatibility option `keepStr = TRUE`.
- The `as.POSIXlt(<POSIXlt>)` and `as.POSIXct(<POSIXct>)` default methods now do obey their `tz` argument, also in this case.
- `as.POSIXlt(<Date>)` now does apply a `tz` (time zone) argument, as does `as.POSIXct()`; partly suggested by Roland Fuß on the R-devel mailing list.
- `as.Date.POSIXlt(x)` now also works when the list components are of unequal length, aka "partially filled" or "ragged".
- `expand.model.frame()` looked up variables in the wrong environment when applied to models fitted without `data`. Reported in PR#18414.
- `time()` now (also) uses the `ts.eps = getOption("ts.eps")` argument and thus by default rounds values very close to the start (or end) of a year. Based on a proposal by Andreï V. Kostyrka on R-help.
- Printing of a `factanal()` result with just one factor and `sort = TRUE` now works regularly, fixing PR#17863 by Timothy Bates, thanks to the 'R Contributors' working group.
- Printing 0-length objects of class `"factor"`, `"roman"`, `"hexmode"`, `"octmode"`, `"person"`, `"bibentry"`, or `"citation"` now prints something better, one of which fixes PR#18422, reported by Benjamin Feakins.
- `Sys.timezone()` queries `timedatectl` only if `systemd` is loaded; addressing a report by Jan Gorecki in PR#17421.
- The formula method of `cor.test()` had scoping problems when `environment(formula)` was not the calling environment; reported with a patch proposal by Mao Kobayashi in PR#18439.
- `attach()` of an environment with active bindings now preserves the active bindings. Reported by Kevin Ushey in PR#18425.
- BLAS detection now works also with system-provided libraries not available as regular files. This fixes detection of the Accelerate framework on macOS since Big Sur. Reported by David Novgorodsky.
- `download.file()` gives a helpful error message in case of an invalid `download.file.method` option, thanks to Colin Fay's report in PR#18455.
- Sporadic crashes of `Rterm` when using completion have been fixed.
- `Rprof()` is now more reliable. A livelock in thread initialization with too short sampling interval has been fixed on macOS. A deadlock in using the C runtime has been fixed on Windows. A potential deadlock has been prevented on Unix.
- Cursor placement in `Rgui` now works even after a fixed-width font is selected.
- Mandatory options (`options()`) are now set on startup so that saving and restoring them always works (PR#18372).
- Package installation, `R CMD INSTALL` or `install.packages(*)`, now parses each of the '`<pkg>/R/*.R`' files individually instead of first concatenating and then `parse()`ing the large resulting file. This allows parser or syntax errors to be diagnosed with correct file names and line numbers, thanks to Simon Dedman's report and Bill Dunlap's patch in PR#17859.
  This *does* require syntactically self contained R source files now, fixing another inadvertent bug.
- `predict.lm(<model with offset>)` now finds the offset in the correct environment, thanks to André Gillibert's report and patch in PR#18456.
- `getInitial(<formula>)` now finds the `selfStart` model in the correct environment. (Reported by Ivan Krylov in PR#18368.)

- Fix for possible segfault when using recently-added graphics features, such as gradients, clipping paths, masks, and groups with `pdf(file=NULL)`.
- `class(m) <- class(m)` no longer changes a matrix `m` by adding a class *attribute*.
- `packageDate(pkg)` now only warns once if there is no `pkg`.
- When `ts()` creates a multivariate time series, `"mts"`, it also inherits from `"array"` now, and `is.mts()` is documented *and* stricter.
- `Rd2txt()` now preserves line breaks of `\verb` Rd content and from duplicated `\cr`. The former also fixes the rendering of verbatim output from Rd `\Sexpr` in plain-text help.
- `uniroot(f, interval)` should no longer wrongly converge *outside* the interval in some cases where `abs(f(x)) == Inf` for an `x` at the interval boundary, thanks to posts by Ben Bolker and Serguei Sokol on R-devel.
- Vectorized alpha handling in palette functions such as in `gray()`, `rainbow()`, or `hcl.colors()` works correctly now, thanks to Achim Zeileis' report and patch in PR#18476.
- Formatting and `print()`ing of `bibentry` objects has dropped the deprecated `citation.bibtex.max` argument, such that the `bibtex` argument's default for `print.bibentry()` depends directly on the `citation.bibtex.max` option, whereas in `format.bibentry()` the option no longer applies.
- Attempting to use a character string naming a foreign function entry point in a foreign function call in a package will now signal an error if the packages has called `R_forceSymbols` to specify that symbols must be used.
- An error in `table()` could permanently set `options(warn=2)` promoting all subsequent warnings to errors.
- The `sigma()` function gave misleading results for binary GLMs. A new method for objects of class `"glm"` returns the square root of the estimate of the dispersion parameter using the same calculation as `summary.glm()`.
- `bs()` and `ns()` in the (typical) case of automatic knot construction, when some of the supposedly inner knots coincide with boundary knots, now moves them inside (with a warning), building on PR#18442 by Ben Bolker.
- `R CMD` on Windows now skips the site profile with '`--no-site-file`' and '`--vanilla`' even when `R_PROFILE` is set (PR#18512, from Kevin Ushey).



Changes in R 4.2.3

C-Level Facilities:

- The definition of `DL_FUNC` in '`R_ext/Rdynload.h`' has been changed to be fully C-compliant. This means that functions loaded *via* for example `R_GetCCallable` need to be cast to an appropriate type if they have any arguments.
- `.Machine` has a new element `sizeof.time_t` to identify old systems with a 32-bit type and hence a limited range of date-times (and limited support for dates millions of years from present).

Package Installation:

- (Windows) The default C++ standard had accidentally been left at C++11 when it was changed to C++14 on Unix.

Bug Fixes:

- As `"POSIXlt"` objects may be "partially filled" and their list components meant to be recycled, `length()` now is the length of the longest component.
- `as.POSIXlt.Date()` could underflow for dates in the far past (more than half a million years BCE).
- `as.Date.POSIXlt(x)` would return `"1970-01-01"` instead of `NA` in R 4.2.2, e.g., for
```
x <- as.POSIXlt(c("2019-01-30","2001-1-1"))
x$mon <- c(0L, NA); as.Date(x)
```
- `R CMD check` failed to apply enabled `_R_CHECK_SUGGESTS_ONLY_` to examples and vignettes (regression in R 4.2.0).
- `R CMD check` did not re-build vignettes in separate processes by default (regression in R 4.2.0).
- Running examples from HTML documentation now restores previous **knitr** settings and options (PR#18420).
- Quartz: fonts are now located using Core Graphics API instead of deprecated ATS which is no longer supported in the macOS 13 SDK (PR#18426). This also addresses an issue where the currently used font in the Quartz device context was not correctly retained.
- (Windows) Math symbols in text drawing functions are again rendered correctly (PR#18440). This fixes a regression in R 4.2.1 caused by a fix in PR#18382 which uncovered an issue in GraphApp due to which the symbol charset was not used with TT Symbol font face.
- (Windows) Installing a package with a '`src/Makefile.{win,ucrt}`' file includes '`~/.R/Makevars.win64`' in the search for user makevars, as documented in "R Installation and Administration" and done for packages with a '`src/Makevars.{win,ucrt}`' file.
- `format(<POSIXlt_w/_unbalanced_sec>, "....%OS<n>")` with $n > 0$ no longer accidentally uses the unbalanced seconds, thanks to Suharto Anggono's report (including patch) in PR#18448.
- `solve.default(a, b)` works around issues with some versions of LAPACK when `a` contains `NA` or `NaN` values.
- When `UseMethod()` cannot dispatch, it no longer segfaults producing the error message in case of a long `class()`, thanks to Joris Vankerschaver's report (including patch) in PR#18447.
- When `example(foo, ..)` produces graphics on an interactive device it needs to open itself, it now leaves `devAskNewPage()` unchanged even when it was `FALSE`, thus fixing a 14 years old 'FIXME'.
- `packageDescription()` again catches errors from encoding conversions. This also fixes broken `packageVersion()` in C locale on systems where `iconv` does not support transliteration.



Changes in R 4.2.2

New Features:

- `tools::Rdiff(useDiff = TRUE)` checks for the presence of an external `diff` command and switches to `useDiff = FALSE` if none is found. This allows `R CMD Rdiff` to always work.
- On Windows, environment variable `R_LIBCURL_SSL_REVOKE_BEST_EFFORT` can be used to switch to only 'best-effort' SSL certificate revocation checks with the default `"libcurl"` download method. This reduces security, but may be needed for downloads to work with MITM proxies (PR#18379).

- (macOS) The run-time check for libraries from XQuartz for X11 and Tcl/Tk no longer uses `otool` from the Apple Developer Tools (PR#18400).
- The LaTeX style for producing the PDF manuals, '`Rd.sty`', now loads the standard '`amsmath`', '`amsfonts`' and '`amssymb`' packages for greater coverage of math commands in the Rd `\eqn` and `\deqn` macros. The `\mathscr` LaTeX command is also provided (via the '`mathrsfs`' package, if available, or the '`amsfonts`' bundle otherwise), fulfilling the wish of PR#18398.
- (Windows) The default format of `readClipboard()` and `writeClipboard()` has been changed to `13` (`CF_UNICODETEXT`).

Installation on a Unix-Alike:

- The PDF manuals (if built) can be compacted by the new target `make compact-pdf` (at the top level or in directory '`doc/manual`').
- There is now `configure` support for LLVM clang 15 on Linux, which defaults to position-independent (PIE) executables whereas `gfortran` does not.
- Many small changes to ease compilation (and suppress warnings) with LLVM `clang` 15.

Bug Fixes:

- `Rscript -e` would fail if '`stdin`' were closed (Reported by Henrik Bengtsson.)
- `qt(*, log.p=TRUE)` in outer tails no longer produces `NaN` in its final steps, thus fixing PR#18360.
- `tools::Rd2latex()` now escapes hashes and ampersands when writing URLs, fixing LaTeX errors with such URLs in `\tabular`.
- When `isGeneric(f, fdef=*)` is used with mismatching names, the warning is better understandable; reported (with fix) in PR#18370 by Gabe Becker.
- `poly(x, n)` now works again (and is now documented) when `x` is a `"Date"` or `"POSIXct"` object, or of another class while fulfilling `mode(x) == "numeric"`. This also enables `poly(x, *, raw=TRUE)` for such variables. Reported by Michael Chirico to R-devel.
- `write.table()`, `write.csv()` and `write.csv2()` restore their numerical precision (internal equivalent of `digits = 15`) after an interrupt (PR#18384).
- One can now read also byte `FF` from a clipboard connection (PR#18385).
- `source("")` and `source(character())` now give more helpful error messages.
- `R CMD check --as-cran` set `_R_CHECK_TIMINGS_` too late to have the intended effect.
- `as.POSIXlt(x)` now also works with very large dates `x`, fixing PR#18401 reported by Hannes Mühleisen.
- Files can now be extracted even from very large zip archives (PR#18390, thanks to Martin Jakt).
- Non-finite objects of class `"POSIXlt"` are now correctly coerced to classes `"Date"` and `"POSIXct"`; following up on the extension to `format()` them correctly.
- Added methods for `is.finite()`, `is.infinite()` and `is.nan()` for `"POSIXlt"` date-time objects.

Bug Fixes (Windows):

- Non-ASCII characters are now properly displayed on Windows in windows created using GraphApp via e.g. `winDialogString` thanks to a workaround for an at least surprising Windows behavior with UTF-8 as the system encoding (PR#18382).
- Find and replace operations work again in the script editor in `Rgui` on Windows.

- Computation of window size based on requested client size in GraphApp when running in a multi-byte locale on Windows has been fixed (regression in R 4.2.0 for users of systems where R 4.1 used a single-byte locale). `Rgui` again respects the number of console rows and columns given in '`Rconsole`' file.
- `Rterm` support for `Alt+xxx` sequences has been fixed to produce the corresponding character (only) once. This fixes pasting text with tilde on Italian keyboard (PR#18391).



Changes in R 4.2.1

New Features:

- New function `utils::findCRANmirror()` to find out if a CRAN mirror has been selected, otherwise fallback to the main site. This behaves in the same way as `tools::CRAN_package_db()` and is intended for packages wishing to access CRAN for purposes other than installing packages.
  The need for this was shown by a day when the main CRAN website was offline and a dozen or so packages which had its URL hardcoded failed their checks.

Installation on a Unix-Alike:

- The libraries searched for by '`--with-blas`' (without a value) now include BLIS (after OpenBLAS but before ATLAS). And on macOS, the Accelerate framework (after ATLAS). (This is patterned after the `AX_BLAS` macro from the Autoconf Archive.)
- The included LAPACK sources have been updated to 3.10.1.

Utilities:

- The (full path to) the command `tidy` to be used for HTML validation can be set by environment variable `R_TIDYCMD`.
- Setting environment variable `_R_CHECK_RD_VALIDATE_RD2HTML_` to a false value will override `R CMD check --as-cran` and turn off HTML validation. This provides a way to circumvent a problematic `tidy`.
  The 2006 version that ships with macOS is always skipped.

C-Level Facilities:

- The undocumented legacy declarations of `Sint`, `Sfloat`, `SINT_MAX` and `SINT_MIN` in header '`R.h`' are deprecated.

Bug Fixes:

- `fisher.test(d)` no longer segfaults for "large" `d`; fixing PR#18336 by preventing/detecting an integer overflow reliably.
- `tar(., files=*)` now produces correctly the warning about invalid UID or GID of files, fixing PR#18344, reported by Martin Morgan.
- `tk_choose.files()` with `multi = FALSE` misbehaved on paths containing spaces (PR#18334) (regression introduced in R 4.0.0).
- `sort(x, partial = ind, *)` now works correctly notably for the non-default `na.last = FALSE` or `TRUE`, fixing PR#18335 reported by James Edwards.
- Environment variable `_R_CHECK_XREFS_REPOSITORIES_` is only used for checking '`.Rd`' cross-references in `R CMD check` (as documented) and not for other uses looking for a CRAN mirror.

- The search for a CRAN mirror when checking packages now uses `getOption("repos")` if that specifies a CRAN mirror, even when it does not also specify all three Bioconductor repositories (as was previously required).
- The HTML code generated by `tools::Rd2HTML()` has been improved to pass `tidy` 5.8.0.

Bug Fixes (Windows):

- Writing to a clipboard connection works again, fixing a regression in R 4.2.0 (PR#18332). Re-using a closed clipboard connection no longer issues a spurious warning about an ignored encoding argument.
- C function `getlocale` no longer attempts to query an unsupported category from the OS, even when requested at R level, which may cause crashes when R 4.2.0 (which uses UCRT) is embedded (reported by Kevin Ushey).
- Accent keys now work in GraphApp Unicode windows, which are used by `Rgui` whenever running in a multibyte locale (so also in UTF-8, hence fixing a regression in R 4.2.0 for users of systems where R 4.1 used a single-byte locale).
- Completion in `Rgui` now works also with non-ASCII characters.
- `Rgui` no longer truncates usage information with '`--help`'.
- Text injection from external applications via `SendInput` now works in GraphApp Unicode windows, fixing a regression in R 4.2.0 for `Rgui` users of systems where R 4.1 used a single-byte locale but R 4.2.0 uses UTF-8.
- Performance of `txtProgressBar()` in `Rgui` when running in a multi-byte locale has been improved (fixing a performance regression in R 4.2.0 for users of systems where R 4.1 used a single-byte locale).
- The script editor in `Rgui` now works also on systems using UTF-8 as the native encoding. Users of the script editor have to convert their scripts with non-ASCII characters to UTF-8 before reading them in R 4.2.1 or newer (on recent Windows where UTF-8 is used). This fixes a regression in R 4.2.0, which prevented some operations with scripts when they contained non-ASCII characters.



Changes in R 4.2.0

Significant User-Visible Changes:

- The `formula` method of `aggregate()` now matches the generic in naming its first argument `x` (resolving PR#18299 by Thomas Soeiro).
  This means that calling `aggregate()` with a formula as a named first argument requires name `formula` in earlier versions of R and name `x` now, so portable code should not name the argument (code in many packages did).
- Calling `&&` or `||` with either argument of length greater than one now gives a warning (which it is intended will become an error).
- Calling `if()` or `while()` with a condition of length greater than one gives an error rather than a warning. Consequently, environment variable `_R_CHECK_LENGTH_1_CONDITION_` no longer has any effect.
- Windows users should consult the Windows section below for some profound changes including
  - Support for 32-bit builds has been dropped.
  - UTF-8 locales are used where available.
  - The default locations for the R installation and personal library folder have been changed.

Thanks to Tomas Kalibera for months of work on the Windows port for this release.

New Features:

- `matrix(x, n, m)` now warns in more cases where `length(x)` differs from `n * m`, as suggested by Abby Spurdle and Wolfgang Huber in Feb 2021 on the R-devel mailing list.
  This warning can be turned into an error by setting environment variable `_R_CHECK_MATRIX_DATA_` to 'TRUE': `R CMD check --as-cran` does so unless it is already set.
- Function `file_test()` in package **utils** gains tests for symlinks, readability and writability.
- `capabilities("libxml")` is now false.
  The description of `capabilities("http/ftp")` now reflects that it refers to the default method, no longer the internal one.
- `simplify2array()` gains an `except` argument for controlling the exceptions used by `sapply()`.
- Environment variables `R_LIBS_USER` and `R_LIBS_SITE` are both now set to the R system default if unset or empty, and can be set to `NULL` to indicate an empty list of user or site library directories.
- The warning for `axis()`(-like) calls in cases of relatively small ranges (typically in log-scale situations) is slightly improved *and* suppressed from explicit calls to `.axisPars()` as has always been the intention.
- The `contrasts` setter function `` `contrasts<-` `` gains an explicit default `how.many = NULL` rather than just using `missing(how.many)`.
- `grid.pretty()` gains a new optional argument `n = 5`.
- There is a new function `.pretty()` with option `bounds` as a technical-utility version of `pretty()`. It and `pretty()` gain a new argument `f.min` with a better than back-compatible default.
- Function `grDevices::axisTicks()` and related functions such as `graphics::axis()` work better, notably for the log scale; partly because of the `pretty()` improvements, but also because care is taken e.g., when `ylim` is finite but `diff(ylim)` is infinite.
- `nclass.FD()` gains a `digits` option.
- The R Mathlib internal C function `bd0()` (called indirectly from a dozen probability density and distribution functions such as `dpois()`, `dbinom()`, `dgamma()`, `pgamma()` *etc*) has been complemented by a more sophisticated and (mostly) more accurate C function `ebd0()`, currently called only by internal `dpois_raw()` improving accuracy for R level `dpois()` and potentially others calling it such as `dnbinom()`, `dgamma()` or `pgamma()`. (Thanks to Morten Welinder's PR#15628.)
- `write.ftable()` gains `sep = " "` argument as suggested by Thomas Soeiro.
- The names of the locale categories supported by R's `Sys.getlocale()` and `Sys.setlocale()` are now provided by variable `.LC.categories` in the `base` namespace.
- The `Date` and `POSIXt` methods for `hist()` and the `histogram` method for `plot()` now also use the new default `col = "lightgray"` in consistency with the corresponding change to `hist()`'s default for R 4.0.0.
- `hist.default()` gains new `fuzz` argument, and the histogram `plot` method no longer uses fractional axis ticks when displaying counts (`"Frequency"`).
- `mapply()` and hence `Map()` now also obey the "max-or-0-if-any" recycling rule, such that, e.g., `Map(`+`, 1:3, 1[0])` is valid now.

- `as.character(<obj>)` for `"hexmode"` or `"octmode"` objects now fulfils the important basic rule `as.character(x)[j] === as.character(x[j])`.
- The set utility functions, notably `intersect()` have been tweaked to be more consistent and symmetric in their two set arguments, also preserving a common `mode`.
- `substr(ch, start,end) <- new` now e.g., preserves `names(ch)`; ditto for `substring()`, thanks to a patch from Brodie Gaslam.
- `plot(<lm>)` gains a `extend.ylim.f` argument, in partial response to PR#15285; further PR#17784 is fixed thanks to several contributors and a patch by Elin Waring. The Cook's dist contours get customizable via `cook.col` and `cook.lty` with a different default color and their legend is nicer by default and customizable via `cook.legendChanges`.
- Attempting to subset an object that is not subsettable now signals an error of class `notSubsettableError`. The non-subsettable object is contained in the `object` field of the error condition.
- Subscript-out-of-bounds errors are now signaled as errors of class `subscriptOutOfBoundsError`.
- Stack-overflow errors are now signaled as errors inheriting from class `stackOverflowError`. See `?stackOverflowError` for more details.
- New partly experimental `Sys.setLanguage()` utility, solving the main problem of PR#18055.
- `gettext()` and `gettextf()` get a new option `trim = TRUE` which when set to false allows translations for strings such as `"Execution halted\n"` typical for C code.
- An experimental implementation of hash tables is now available. See `?hashtab` for more details.
- `identical()` gains a `extptr.as.ref` argument for requesting that external pointer objects be compared as reference objects.
- `reorder()` gets an argument `decreasing` which it passes to `sort()` for level creation; based on the wish and patch by Thomas Soeiro in PR#18243.
- `as.vector()` gains a `data.frame` method which returns a simple named list, also clearing a long-standing 'FIXME' to enable `as.vector(<data.frame>, mode="list")`. This breaks code relying on `as.vector(<data.frame>)` to return the unchanged data frame.
- `legend()` is now vectorized for arguments `cex`, `x.intersp`, and `text.width`. The latter can now also be specified as a vector (one element for each column of the legend) or as `NA` for computing a proper column wise maximum value of `strwidth(legend)`. The argument `y.intersp` can be specified as a vector with one entry for each row of the legend.
  `legend()` also gains new arguments `title.cex` and `title.font`. Thanks to Swetlana Herbrandt.
- Deparsing no longer remaps attribute names `dim`, `dimnames`, `levels`, `names` and `tsp` to historical S-compatible names (which `structure()` maps back).
- `sample()` and `sample.int()` have additional sanity checks on their `size` and `n` arguments.
  `all.equal.numeric()` gains a sanity check on its `tolerance` argument – calling `all.equal(a, b, c)` for three numeric vectors is a surprisingly common error.
  `mean(na.rm =)`, `rank(na.last =)`, `barplot(legend.text =)`, `boxplot()`, `contour(drawlabels =)`, `polygon(border =)` and `methods::is(class2 =)` have more robust sanity checks on their arguments.
  `R CMD Rd2pdf` (used by `R CMD check`) has a more robust sanity check on the format of `\alias{}` commands.

- `psigamma(x, deriv)` for negative `x` now also works for `deriv = 4` and `5`; their underlying C level `dpsifn()` is documented in 'Writing R Extensions'.
- The HTML help system now uses HTML5 (wish of PR#18149).
- `ks.test()` now provides exact p-values also with ties and MC p-values in the two-sample (Smirnov) case. By Torsten Hothorn.
- `ks.test()` gains a formula interface, with `y ~ 1` for the one-sample (Kolmogorov) test and `y ~ group` for the two-sample (Smirnov) test. Contributed by Torsten Hothorn.
- The return value from `ks.test()` now has class `c("ks.test", "htest")` – packages using `try()` need to take care to use `inherits()` and not `==` on the class.
- New functions `psmirnov()`, `qsmirnov()` and `rsmirnov()` in package **stats** implementing the asymptotic and exact distributions of the two-sample Smirnov statistic.
- `iconv()` now allows `sub = "c99"` to use C99-style escapes for UTF-8 inputs which cannot be converted to encoding `to`.
- In a forward pipe `|>` expression it is now possible to use a named argument with the placeholder `_` in the `rhs` call to specify where the `lhs` is to be inserted. The placeholder can only appear once on the `rhs`.
- The included LAPACK sources have been updated to version 3.10.0, except for the four Fortran 77 routines which 3.10.0 has re-implemented in Fortran 90 (where the older versions have been retained as the R build process does not support Fortran 90).
- `path.expand()` and most other uses of tilde expansion now warn if a path would be too long if expanded. (An exception is `file.exists()`, which silently returns false.)
- `trunc(<Date>, *)` now supports `units = "months"` or `"years"` for consistency with the `POSIXt` method, thanks to Dirk Eddelbuettel's proposal in PR#18099.
- `list2DF()` now checks that its arguments are of the same length, rather than use recycling.
- The HTML help system has several new features: LaTeX-like math can be typeset using either KaTeX or MathJax, usage and example code is highlighted using Prism, and for dynamic help the output of examples and demos can be shown within the browser if the **knitr** package is installed. These features can be disabled by setting the environment variable `_R_HELP_ENABLE_ENHANCED_HTML_` to a false value.

Graphics:

- The graphics engine version, `R_GE_version`, has been bumped to `15` and so packages that provide graphics devices should be reinstalled.
- The **grid** package now allows the user to specify a "vector" of pattern fills. The `fill` argument to `gpar()` accepts a list of gradients and/or patterns and the functions `linearGradient()`, `radialGradient()`, and `pattern()` have a new `group` argument. Points grobs (data symbols) can now also have a pattern fill.
  The `grobCoords()` function now returns a more informative and complex result.
- The **grid** package has new functions for drawing isolated groups: `grid.group()`, `grid.define()`, and `grid.use()`. These functions add compositing operators and affine transformations to R's graphics capabilities.
  The **grid** package also has new functions for stroking and filling paths: `grid.stroke()`, `grid.fill()`, and `grid.fillStroke()`.
  A new function `as.path()` allows the user to specify the fill rule for a path that is to be used for clipping, stroking, or filling; available options are `"winding"` and `"evenodd"`. A new function `as.mask()` allows the user to specify the type of a mask; available options are `"alpha"` and `"luminance"`.
  These new features are only supported so far (at most) on the Cairo-based graphics devices and on the `pdf()` device.

- `dev.capabilities()` reports on device support for the new features.
- `par()` now warns about unnamed non-character arguments to prevent misuse such as `{usr <- par("usr"); par(usr)}`.

Windows:

- R uses UTF-8 as the native encoding on recent Windows systems (at least Windows 10 version 1903, Windows Server 2022 or Windows Server 1903). As a part of this change, R uses UCRT as the C runtime. UCRT should be installed manually on systems older than Windows 10 or Windows Server 2016 before installing R.
- The default personal library on Windows, folder '`R\win-library\x.y`' where '`x.y`' stands for R release '`x.y.z`', is now a subdirectory of Local Application Data directory (usually a hidden directory '`C:\Users\username\AppData\Local`'). Use `shell.exec(.libPaths()[1])` from R to open the personal library in Explorer when it is first in the list (PR#17842).
- R uses a new 64-bit Tcl/Tk bundle. The previous 32-bit/64-bit bundle had a different layout and can no longer be used.
- Make files and installer scripts for Windows have been tailored to '`Rtools42`', the newly recommended 64-bit `gcc` 10.3 MinGW-W64 UCRT toolchain.
- '`Rtools42`' by default uses the Windows security features ASLR and DEP; hence CRAN builds of R and packages also do.
- R now supports files '`Makevars.ucrt`', '`Makefile.ucrt`', '`configure.ucrt`' and '`cleanup.ucrt`' in packages, which are used in preference to the '`.win`' variants. This allows keeping the '`.win`' files around to support older versions of R. This feature will be removed in the future once support for older versions of R would no longer be needed.
- `R.version` gains a new field `crt` (only on Windows) to denote the C runtime. The value is `"ucrt"`.
- On Windows, `download.file(method = "auto")` and `url(method = "default")` now follow Unix in using `"libcurl"` for all except '`file://`' URIs.
- '`Rtools42`' includes an unpatched Msys2 build of GNU `tar`. Paths including drive letters can be made to work by adding '`--force-local`' to environment variable `TAR_OPTIONS`. ('`Rtools40`' and earlier included a patched version which defaulted to this option.)
- Installer builds of R automatically find the '`Rtools42`' software collection as well as the compiler toolchain. No `PATH` setting is required from the user.
- The default installation directory of R for a user-only installation has been changed to the User Program Files directory (usually a hidden directory '`C:\Users\username\AppData\Local\Programs`') to follow Windows conventions. Use `shell.exec(R.home())` from R to open the R installation directory in Explorer (PR#17842).
- R now supports installation-time patching of packages. Patches may be installed from a supplied URL or a local directory or disabled. Patches are included into the installed packages for reference. This experimental feature may be removed in the future.
- `libcurl` is now required for building from source.
- The clipboard connection now works also with text in other than the current native encoding (PR#18267, with Hiroaki Yutani). Text is always pasted to the clipboard in UTF16-LE and the `encoding` argument is ignored.
- The internal case-changing functions are now used by default on Windows – this circumvents problems (for example with E acute) of the UCRT Windows' runtime.

- R on Windows now uses the system memory allocator. Doug Lea's allocator was used since R 1.2.0 to mitigate performance limitations seen with system allocators on earlier versions of Windows.
- `memory.limit()` and `memory.size()` are now stubs on Windows (as on Unix-alikes).
- Applications embedding R on Windows can now use additional callbacks, which have so far only been available only on Unix (PR#18286).

Installation:

- Facilities for accessing '`ftp://`' sites are no longer tested (except *pro tem* for `curlGetHeaders()`) as modern browsers have removed support.
- R can now be built with '`DEFS = -DSTRICT_R_HEADERS`' .

Package Installation:

- `R CMD INSTALL` no longer tangles vignettes. This completes an `R CMD build` change in R 3.0.0 and affects packages built before R 3.0.2. Such packages should be re-made with `R CMD build` to have the tangled R code of vignettes shipped with the tarball.
- `USE_FC_LEN_T` will become the default: this uses the correct prototypes for Fortran BLAS/LAPACK routines called from C/C++, and requires adjustment of most such calls – see 'Writing R Extensions' §6.6.1. (This has been supported since R 3.6.2.)
- Package installation speed for packages installed with `keep.source` has been improved. This resolve the issue reported by Ofek Shilon in PR#18236.

Utilities:

- `R CMD check` can optionally report files/directories left behind in home, '`/tmp`' (even though `TMPDIR` is set) and other directories. See the "R Internals" manual for details.
- `R CMD check` now reports byte-compilation errors during installation. These are not usually fatal but may result in parts of the package not being byte-compiled.
- `_R_CHECK_DEPENDS_ONLY_` can be applied selectively to examples, tests and/or vignettes in `R CMD check`: see the "R Internals" manual.
- `_R_CHECK_SRC_MINUS_W_IMPLICIT_` now defaults to true: recent versions of Apple `clang` on macOS have made implicit function declarations in C into a compilation error.
- `R CMD check --as-cran` makes use of the environment variable `AUTORECONF`. See the "R Internals" manual §8 for further details.
- `R CMD check --use-valgrind` also uses `valgrind` when re-building vignettes as some non-Sweave vignettes unhelpfully comment out all their code when `R CMD check` runs vignettes.
- Errors in re-building vignettes (unless there are LaTeX errors) are reported by `R CMD check` as '`ERROR`' rather than '`WARNING`' when running vignettes has been skipped (as it frequently is in CRAN checks and by '`--as-cran`').
- `R CMD Rd2pdf` gains a '`--quiet`' option that is used by `R CMD build` when building the PDF package manual.
- `R CMD Rd2pdf` now always runs LaTeX in batch mode, consistent with Texinfo $\geq$ 6.7. The '`--batch`' option is ignored.
- `R CMD build` and `R CMD check` now include the Rd file name and line numbers in the error message of an `\Sexpr` evaluation failure.
- For packages using the `\doi` Rd macro (now an install-time `\Sexpr`) but no other dynamic Rd content, `R CMD build` now produces a smaller tarball and is considerably faster – skipping temporary package installation.

- `R CMD check` can optionally (but included in '`--as-cran`') validate the HTML produced from the packages '`.Rd`' files. See https://blog.r-project.org/2022/04/08/enhancements-to-html-documentation/: this needs a fairly recent version of HTML Tidy to be available.

C-Level Facilities:

- The non-API header '`R_ext/R-ftp-http.h`' is no longer provided, as the entry points it covered are now all defunct.
- A number of non-API declarations and macro definitions have been moved from the installed header '`Rinternals.h`' to the internal header '`Defn.h`'. Packages that only use entry points and definitions documented to be part of the API as specified in 'Writing R Extensions' §6 should not be affected.
- The macro `USE_RINTERNALS` no longer has any effect when compiling package code. Packages which also use `R_NO_REMAP` will need to ensure that the remapped names are used for calls to API functions that were formerly also made available as macros.
- The deprecated legacy S-compatibility macros `PROBLEM`, `MESSAGE`, `ERROR`, `WARN`, `WARNING`, `RECOVER`, ... are no longer defined in '`R_ext/RS.h`' (included by '`R.h`'). Replace these by calls to `Rf_error` and `Rf_warning` (defined in header '`R_ext/Error.h`' included by '`R.h`').
  Header '`R_ext/RS.h`' no longer includes '`R_ext/Error.h`'.
- Header '`R_ext/Constants.h`' (included by '`R.h`') when included from C++ now includes the C++ header '`cfloat`' rather than the C header '`float.h`' (now possible as C++11 is required).
- The legacy S-compatibility macros `DOUBLE_*` in '`R_ext/Constants.h`' (included by '`R.h`') are deprecated.
- The deprecated S-compatibility macros `SINGLE_*` in '`R_ext/Constants.h`' (included by '`R.h`') have been removed.
- `R_Calloc`, `R_Free` and `R_Realloc` are preferred to their unprefixed forms and error messages now use the prefix. These forms were introduced in R 3.4.0 and are available even when `STRICT_R_HEADERS` is defined.
- `rmultinom` has been documented in 'Writing R Extensions' §6 so is now part of the R API.
- Similarly, `Rtanpi`, called from R level `tanpi()` is now part of the R API.
- The long-deprecated, undocumented and non-API entry point `call_R` is no longer declared in '`R_ext/RS.h`' (included by '`R.h`').
- The header '`S.h`' which has been unsupported since Jan 2016 has been removed. Use '`R.h`' instead.

Deprecated and Defunct:

- The (non-default and deprecated) `method = "internal"` for `download.file()` and `url()` no longer supports '`http://`' nor '`ftp://`' URIs. (It is used only for '`file://`' URIs.)
  On Windows, `download.file(method = "wininet")` no longer supports '`ftp://`' URIs. (It is no longer the default method, which is `"libcurl"` and does.)
  On Windows, the deprecated `method = "wininet"` now gives a warning for '`http://`' and '`https://`' URIs for both `download.file()` and `url()`. (It is no longer the default method.)
- On Windows, the command-line option '`--max-mem-size`' and environment variable `R_MAX_MEM_SIZE` are defunct. The memory allocation limit was important for 32-bit builds, but these are no longer supported.

- `default.stringsAsFactors()` is now formally deprecated, where that was only mentioned on its regular help page, previously. So it now gives a warning if called.
- `unix.time()` is defunct now; it had been deprecated since R 3.4.0.

Bug Fixes:

- Setting `digits = 0` in `format()`, `print.default()` (and hence typically `print()`) or `options()` is again invalid. Its behaviour was platform-dependent, and it is unclear what "zero significant digits" should mean (PR#18098).
- Messages from C code in the '`cairo`' section of package **grDevices** are now also offered for translation, thanks to Michael Chirico's PR#18123.
- `mean(x)` with finite `x` now is finite also without "long.double" capability.
- `R CMD Rd2pdf` no longer leaves an empty build directory behind when it aborts due to an already existing output file. (Thanks to Sebastian Meyer's PR#18141.)
- `density(x, weights = w, na.rm = TRUE)` when `anyNA(x)` is true, now removes weights "in parallel" to `x`, fixing PR#18151, reported by Matthias Gondan. Additionally, it gets a `subdensity` option.
- Conversion of `\Sexpr[]{<expR>}` to LaTeX or HTML no longer produces long blocks of empty lines when `<expR>` itself contains several lines all producing empty output. Thanks to a report and patch by Ivan Krylov posted to R-devel.
- `R CMD build` no longer fails if a package vignette uses child documents and '`inst/doc`' exists. (Thanks to Sebastian Meyer's PR#18156.)
- When an R documentation ('help' source) file '`man/foo.Rd`' in a package has `\donttest{..}` examples with a syntax error, it is now signalled as ERROR and with correct line numbers relating to the '`*-Ex.R`' file, thanks to Duncan Murdoch and Sebastian Meyer's reports and patch proposals in PR#17501.
- Improved determination of the correct translation domain in non-base packages, addressing the combination of PR#18092 and PR#17998 ('`#c6`') with reports and *augmented* patch #2904 by Suharto Anggono.
  Note that `"R-base"` is no longer the default domain e.g., for top-level calls to `gettext()`; rather translation needs explicit `domain = *` specification in such cases.
- `identical(attrib.as.set=FALSE)` now works correctly with data frames with default row names. (Thanks to Charlie Gao's PR#18179.)
- `txtProgressBar()` now enforces a non-zero width for argument `char`, without which no progress can be visible.
- `dimnames(table(d))` is more consistent in the case where `d` is a list with a single component, thanks to Thomas Soeiro's report to R-devel.
  Further, `table(d1, d2)` now gives an error when `d1` and `d2` are data frames as suggested by Thomas in PR#18224.
- Fix for drawing semi-transparent lines and fills on the native Windows graphics device (PR#18219 and PR#16694). Thanks to Nick Ray for helpful diagnosis on Bugzilla.
- The deparser now wraps sub-expressions such as `if(A) ..` with parentheses when needed; thanks to Duncan Murdoch's PR#18232 and Lionel Henry's patches there.
- `remove.packages()` no longer tries to uninstall `Priority: base` packages, thanks to a report and suggestions by Colin Fay in PR#18227.
- `win.metafile()` now has `xpinch` and `ypinch` arguments so that the user can override Windows' (potentially wrong) guess at device dimensions.
- `x[i]` and `x[[i]]` for non-integer `i` should now behave in all cases as always documented: the index used is equivalent to `as.integer(i)` unless that would overflow where `trunc(i)` is used instead; thanks to Suharto Anggono's report and patch proposals in PR#17977.

- `asOneSidedFormula()` now associates the resulting formula with the global environment rather than the evaluation environment created for the call.
- `<bibentry>$name` now matches the field name case-insensitively, consistent with `bibentry()` creation and the replacement method.
- `cbind()` failed to detect some length mismatches with a mixture of time-series and non-time-series inputs.
- The default LaTeX style file '`Sweave.sty`' used by the `RweaveLatex` driver no longer loads the obsolete '`ae`' package; thanks to a report by Thomas Soeiro in PR#18271. Furthermore, it now skips '`\usepackage[T1]{fontenc}`' for engines other than pdfTeX (if detected) or if the new '`[nofontenc]`' option is used.
- `smooth.spline()` now stores its logical `cv` argument more safely, fixing a rare bug when printing, and also stores `n`.
- `smooth.spline(x,y,*)` now computes the `cv.crit` statistic correctly, also when `is.unsorted(x)`, fixing PR#18294.
- The `data.frame` method of `rbind()` now warns when binding not-wholly-recycling vectors, by analogy to the default method (for matrices).
- `setAs()` finds the correct class for name `to` when multiple packages define a class with that name. Thanks to Gabor Csardi for the report.
- Fix for detaching a package when two classes of the same name are present in method signatures for the same generic. Thanks to Gabor Csardi for the report.
- `match.arg("", c("", "a", "B"))` gives a better error message, in part from PR#17959, thanks to Elin Waring.
- `R CMD Sweave --clean` no longer removes pre-existing files or subdirectories (PR#18242).
- The `quartz()` device no longer splits polylines into subpaths. That has caused narrowly-spaced lines with many points to always look solid even when dashed line type was used due to dash phase restarts.
- Deparsing constructs such as `quote(1 + `!`(2) + 3)` works again as before R 3.5.0, thanks to the report and patch in PR#18284 by Suharto Anggono.
- `as.list(f)` for a `factor f` now keeps `names(f)`, fixing PR#18309.
- `qbeta(.001, .9, .009)` and analogous `qf()` calls now return a correct value instead of `NaN` or wrongly `1`, all with a warning; thanks to the report by Ludger Goeminne in PR#18302.
- `plot.lm()` failed to produce the plot of residuals vs. factor levels (i.e., `which=5` when leverages are constant) for models with character predictors (PR#17840).
- `interaction.plot(..., xtick = TRUE)` misplaced the x-axis line (PR#18305).
- Not strictly fixing a bug, `format()`ing and `print()`ing of non-finite `Date` and `POSIXt` values `NaN` and ±`Inf` no longer show as `NA` but the respective string, e.g., `Inf`, for consistency with numeric vector's behaviour, fulfilling the wish of PR#18308.
- `R CMD check` no longer runs test scripts generated from corresponding '`.Rin`' files twice and now signals an ERROR if processing an '`.Rin`' script fails.
- `tools::Rd2txt()` used for plain-text help pages now renders `\href`s (if `tools::Rd2txt_options(showURLs = TRUE)`) and `\url`s with percent-encoding and standards-compliant delimiting style (angle brackets and no '`URL: `' prefix). `\email` is now rendered with a '`mailto:`' prefix.



Changes in R 4.1.3

New Features:

- The default version of Bioconductor has been changed to 3.14. (This is used by `setRepositories` and the menus in GUIs.)

Utilities:

- `R CMD check --as-cran` has a workaround for a bug in versions of `file` up to at least 5.41 which mis-identify DBF files last changed in 2022 as executables.

C-Level Facilities:

- The legacy S-compatibility macros `SINGLE_*` in '`R_ext/Constants.h`' (included by '`R.h`') are deprecated and will be removed in R 4.2.0.

Bug Fixes:

- Initialization of self-starting `nls()` models with initialization functions following the pre-R-4.1.0 API (without the `...` argument) works again for now, with a deprecation warning.
- Fixed quoting of `~autodetect~` in Java setting defaults to avoid inadvertent user lookup due to leading `~`, reported in PR#18231 by Harold Gutch.
- `substr(., start, stop) <- v` now treats *negative* `stop` values correctly. Reported with a patch in PR#18228 by Brodie Gaslam.
- Subscripting an array `x` without dimnames by a `length(dim(x))`-column character matrix gave "random" non-sense, now an error; reported in PR#18244 by Mikael Jagan.
- `...names()` now matches `names(list(...))` closely, fixing PR#18247.
- `all.equal(*, scale = s)` now works as intended when `length(s) > 1`, partly thanks to Michael Chirico's PR#18272.
- `print(x)` for long vectors `x` now also works for named atomic vectors or lists and prints the correct number when reaching the `getOption("max.print")` limit; partly thanks to a report and proposal by Hugh Parsonage to the R-devel list.
- `all.equal(<selfStart>, *)` no longer signals a deprecation warning.
- `reformulate(*, response=r)` gives a helpful error message now when `length(r) > 1`, thanks to Bill Dunlap's PR#18281.
- Modifying `globalCallingHandlers` inside `withCallingHandlers()` now works or fails correctly, thanks to Henrik Bengtsson's PR#18257.
- `hist(<Date>, breaks = "days")` and `hist(<POSIXt>, breaks = "secs")` no longer fail for inputs of length 1.
- `qbeta(.001, .9, .009)` and similar cases now converge correctly thanks to Ben Bolker's report in PR#17746.
- `window(x, start, end)` no longer wrongly signals "'start' cannot be after 'end'", fixing PR#17527 and PR#18291.
- `data()` now checks that its (rarely used) `list` argument is a character vector – a couple of packages passed other types and gave incorrect results.
- `which()` now checks its `arr.ind` argument is TRUE rather coercing to logical and taking the first element – which gave incorrect results in package code.
- `model.weights()` and `model.offset()` more carefully extract their model components, thanks to Ben Bolker and Tim Taylor's R-devel post.
- `list.files(recursive = TRUE)` now shows all broken symlinks (previously, some of them may have been omitted, PR#18296).



Changes in R 4.1.2

C-Level Facilities:

- The workaround in headers 'R.h' and 'Rmath.h' (`using namespace std;`) for the Oracle Developer Studio compiler is no longer needed now C++11 is required so has been removed. A couple more usages of `log()` (which should have been `std::log()`) with an `int` argument are reported on Solaris.
- The undocumented limit of 4095 bytes on messages from the S-compatibility macros `PROBLEM` and `MESSAGE` is now documented and longer messages will be silently truncated rather than potentially causing segfaults.
- If the `R_NO_SEGV_HANDLER` environment variable is non-empty, the signal handler for SEGV/ILL/BUS signals (which offers recovery user interface) is not set. This allows more reliable debugging of crashes that involve the console.

Deprecated and Defunct:

- The legacy S-compatibility macros `PROBLEM`, `MESSAGE`, `ERROR`, `WARN`, `WARNING`, `RECOVER`, ... are deprecated and will be hidden in R 4.2.0. R's native interface of `Rf_error` and `Rf_warning` has long been preferred.

Bug Fixes:

- `.mapply(F, dots, .)` no longer segfaults when `dots` is not a `list` and uses `match.fun(F)` as always documented; reported by Andrew Simmons in PR#18164.
- `hist(<Date>, ...)` and `hist(<POSIXt>, ...)` no longer pass arguments for `rect()` (such as `col` and `density`) to `axis()`. (Thanks to Sebastian Meyer's PR#18171.)
- `\Sexpr{ch}` now preserves `Encoding(ch)`. (Thanks to report and patch by Jeroen Ooms in PR#18152.)
- Setting the RNG to `"Marsaglia-Multicarry"` e.g., by `RNGkind()`, now warns in more places, thanks to André Gillibert's report and patch in PR#18168.
- `gray(numeric(), alpha=1/2)` no longer segfaults, fixing PR#18183, reported by Till Krenz.
- Fixed `dnbinom(x, size=<very_small>, .., log=TRUE)` regression, reported by Martin Morgan.
- `as.Date.POSIXlt(x)` now keeps `names(x)`, thanks to Davis Vaughan's report and patch in PR#18188.
- `model.response()` now strips an `"AsIs"` class typically, thanks to Duncan Murdoch's report and other discussants in PR#18190.
- `try()` is considerably faster in case of an error and long call, as e.g., from some `do.call()`. Thanks to Alexander Kaever's suggestion posted to R-devel.
- `qqline(y = <object>)` such as `y=I(.)`, now works, see also PR#18190.
- Non-integer `mgp par()` settings are now handled correctly in `axis()` and `mtext()`, thanks to Mikael Jagan and Duncan Murdoch's report and suggestion in PR#18194.
- `formatC(x)` returns length zero `character()` now, rather than `""` when `x` is of length zero, as documented, thanks to Davis Vaughan's post to R-devel.
- `removeSource(fn)` now retains (other) `attributes(fn)`.



Changes in R 4.1.1

New Features:

- `require(`*pkg*`, quietly = TRUE)` is quieter and in particular does not warn if the package is not found.

Deprecated and Defunct:

- Use of '`ftp://`' URIs should be regarded as deprecated, with on-going support confined to `method = "libcurl"` and not routinely tested. (Nowadays no major browser supports them.)
- The non-default `method = "internal"` is deprecated for '`http://`' and '`ftp://`' URIs for both `download.file` and `url`.
- On Windows, `method = "wininet"` is deprecated for '`http://`', '`https://`' and '`ftp://`' URIs for both `download.file` and `url`. (A warning is only given for '`ftp://`'.)
  For '`ftp://`' URIs the default method is now `"libcurl"` if available (which it is on CRAN builds).
  `method = "wininet"` remains the default for '`http://`' and '`https://`' URIs but if `libcurl` is available, using `method = "libcurl"` is preferred.

Installation:

- `make check` now works also without a LaTeX installation. (Thanks to Sebastian Meyer's PR#18103.)

Bug Fixes:

- `make check-devel` works again in an R build configured with '`--without-recommended-packages`'.
- `qnbinom(p, size, mu)` for large `size/mu` is correct now in a range of cases (PR#18095); similarly for the `(size, prob)` parametrization of the negative binomial. Also `qpois()` and `qbinom()` are better and or faster for extreme cases. The underlying C code has been modularized and is common to all four cases of discrete distributions.
- `gap.axis` is now part of the `axis()` arguments which are passed from `bxp()`, and hence `boxplot()`. (Thanks to Martin Smith's report and suggestions in PR#18109.)
- `.First` and `.Last` can again be set from the site profile.
- `seq.int(from, to, *)` and `seq.default(..)` now work better in large range cases where `from-to` is infinite where the two boundaries are finite.
- `all.equal(x,y)` now returns `TRUE` correctly also when several entries of `abs(x)` and `abs(y)` are close to `.Machine$double.xmax`, the largest finite `numeric`.
- `model.frame()` now clears the object bit when removing the `class` attribute of a value via `na.action` (PR#18100).
- `charClass()` now works with multi-character strings on Windows (PR#18104, fixed by Bill Dunlap).
- `encodeString()` on Solaris now works again in Latin-1 encoding on characters represented differently in UTF-8. Support for surrogate pairs on Solaris has been improved.
- `file.show()` on Windows now works with non-ASCII path names representable in the current native encoding (PR#18132).
- Embedded R on Windows can now find R home directory via the registry even when installed only for the current user (PR#18135).
- `pretty(x)` with finite `x` now returns finite values also in the case where the extreme `x` values are close in size to the maximal representable number `.Machine$double.xmax`. Also, it's been tweaked for very small ranges and when a boundary is close (or equal) to zero; e.g., `pretty(c(0,1e-317))` no longer has negative numbers, currently still warning about a very small range, and `pretty(2^-(1024 - 2^-1/(c(24,10))))` is more accurate.

- The error message for not finding vignette files when weaving has correct file sizes now. (Thanks to Sebastian Meyer's PR#18154.)
- `dnbinom(20, <large>, 1)` now correctly gives 0, and similar cases are more accurate with underflow precaution. (Reported by Francisco Vera Alcivar in PR#18072.)



Changes in R 4.1.0

Future Directions:

- It is planned that the 4.1.x series will be the last to support 32-bit Windows, with production of binary packages for that series continuing until early 2023.

Significant User-Visible Changes:

- Data set `esoph` in package **datasets** now provides the correct numbers of controls; previously it had the numbers of cases added to these. (Reported by Alexander Fowler in PR#17964.)

New Features:

- '`www.omegahat.net`' is no longer one of the repositories known by default to `setRepositories()`. (Nowadays it only provides source packages and is often unavailable.)
- Function `package_dependencies()` (in package **tools**) can now use different dependency types for direct and recursive dependencies.
- The checking of the size of tarball in `R CMD check --as-cran <pkg>` may be tweaked via the new environment variable `_R_CHECK_CRAN_INCOMING_TARBALL_THRESHOLD_`, as suggested in PR#17777 by Jan Gorecki.
- Using `c()` to combine a factor with other factors now gives a factor, an ordered factor when combining ordered factors with identical levels.
- `apply()` gains a `simplify` argument to allow disabling of simplification of results.
- The `format()` method for class `"ftable"` gets a new option `justify`. (Suggested by Thomas Soeiro.)
- New `...names()` utility. (Proposed by Neal Fultz in PR#17705.)
- `type.convert()` now warns when its `as.is` argument is not specified, as the help file always said it *should*. In that case, the default is changed to `TRUE` in line with its change in `read.table()` (related to `stringsAsFactors`) in R 4.0.0.
- When printing list arrays, classed objects are now shown *via* their `format()` value if this is a short enough character string, or by giving the first elements of their class vector and their length.
- `capabilities()` gets new entry `"Rprof"` which is `TRUE` when R has been configured with the equivalent of `--enable-R-profiling` (as it is by default). (Related to Michael Orlitzky's report PR#17836.)
- `str(xS4)` now also shows extraneous attributes of an S4 object `xS4`.
- Rudimentary support for vi-style tags in `rtags()` and `R CMD rtags` has been added. (Based on a patch from Neal Fultz in PR#17214.)
- `checkRdContents()` is now exported from **tools**; it and also `checkDocFiles()` have a new option `chkInternal` allowing to check Rd files marked with keyword `"internal"` as well. The latter can be activated for `R CMD check` via environment variable `_R_CHECK_RD_INTERNAL_TOO_`.
- New functions `numToBits()` and `numToInts()` extend the `raw` conversion utilities to (double precision) `numeric`.

- Functions `URLencode()` and `URLdecode()` in package **utils** now work on vectors of URIs. (Based on patch from Bob Rudis submitted with PR#17873.)
- `path.expand()` can expand '`~user`' on most Unix-alikes even when `readline` is not in use. It tries harder to expand '`~`', for example should environment variable `HOME` be unset.
- For HTML help (both dynamic and static), Rd file links to help pages in external packages are now treated as references to topics rather than file names, and fall back to a file link only if the topic is not found in the target package. The earlier rule which prioritized file names over topics can be restored by setting the environment variable `_R_HELP_LINKS_TO_TOPICS_` to a false value.
- `c()` now removes `NULL` arguments before dispatching to methods, thus simplifying the implementation of `c()` methods, *but* for back compatibility keeps `NULL` when it is the first argument. (From a report and patch proposal by Lionel Henry in PR#17900.)
- `Vectorize()`'s result function's environment no longer keeps unneeded objects.
- Function `...elt()` now propagates visibility consistently with `..n`. (Thanks to Lionel Henry's PR#17905.)
- `capture.output()` no longer uses non-standard evaluation to evaluate its arguments. This makes evaluation of functions like `parent.frame()` more consistent. (Thanks to Lionel Henry's PR#17907.)
- `packBits(bits, type="double")` now works as inverse of `numToBits()`. (Thanks to Bill Dunlap's proposal in PR#17914.)
- `curlGetHeaders()` has two new arguments, `timeout` to specify the timeout for that call (overriding `getOption("timeout")`) and `TLS` to specify the minimum TLS protocol version to be used for `https://` URIs (*inter alia* providing a means to check for sites using deprecated TLS versions 1.0 and 1.1).
- For `nls()`, an optional constant `scaleOffset` may be added to the denominator of the relative offset convergence test for cases where the fit of a model is expected to be exact, thanks to a proposal by John Nash. `nls(*, trace=TRUE)` now also shows the convergence criterion.
- Numeric differentiation *via* `numericDeriv()` gets new optional arguments `eps` and `central`, the latter for taking central divided differences. The latter can be activated for `nls()` via `nls.control(nDcentral = TRUE)`.
- `nls()` now passes the `trace` and `control` arguments to `getInitial()`, notably for all self-starting models, so these can also be fit in zero-noise situations via a `scaleOffset`. For this reason, the `initial` function of a `selfStart` model must now have `...` in its argument list.
- `bquote(splice = TRUE)` can now splice expression vectors with attributes: this makes it possible to splice the result of `parse(keep.source = TRUE)`. (Report and patch provided by Lionel Henry in PR#17869.)
- `textConnection()` gets an optional `name` argument.
- `get()`, `exists()`, and `get0()` now signal an error if the first argument has length greater than 1. Previously additional elements were silently ignored. (Suggested by Antoine Fabri on R-devel.)
- R now provides a shorthand notation for creating functions, e.g. `\(x) x + 1` is parsed as `function(x) x + 1`.
- R now provides a simple native forward pipe syntax `|>`. The simple form of the forward pipe inserts the left-hand side as the first argument in the right-hand side call. The pipe implementation as a syntax transformation was motivated by suggestions from Jim Hester and Lionel Henry.

- `all.equal(f, g)` for `functions` now by default also compares their `environment(.)`s, notably via new `all.equal` method for class `function`. Comparison of `nls()` fits, e.g., may now need `all.equal(m1, m2,check.environment = FALSE)`.
- `.libPaths()` gets a new option `include.site`, allowing to *not* include the site library. (Thanks to Dario Strbenac's suggestion and Gabe Becker's PR#18016.)
- Lithuanian translations are now available. (Thanks to Rimantas Žakauskas.)
- `names()` now works for `DOTSXP` objects. On the other hand, in 'R-lang', the R language manual, we now warn against relying on the structure or even existence of such dot-dot-dot objects.
- `all.equal()` no longer gives an error on `DOTSXP` objects.
- `capabilities("cairo")` now applies only to the file-based devices as it is now possible (if very unusual) to build R with Cairo support for those but not for `X11()`.
- There is optional support for tracing the progress of `loadNamespace()` — see its help.
- (Not Windows.) `l10n_info()` reports an additional element, the name of the encoding as reported by the OS (which may differ from the encoding part (if any) of the result from `Sys.getlocale("LC_CTYPE")`).
- New function `gregexec()` which generalizes `regexec()` to find *all* disjoint matches and all substrings corresponding to parenthesized subexpressions of the given regular expression. (Contributed by Brodie Gaslam.)
- New function `charClass()` in package **utils** to query the wide-character classification functions in use (such as `iswprint`).
- The names of `quantile()`'s result no longer depend on the global `getOption("digits")`, but `quantile()` gets a new optional argument `digits = 7` instead.
- `grep()`, `sub()`, `regexp` and variants work considerably faster for long factors with few levels. (Thanks to Michael Chirico's PR#18063.)
- Provide grouping of `x11()` graphics windows within a window manager such as `Gnome` or `Unity`; thanks to a patch by Ivan Krylov posted to R-devel.
- The `split()` method for class `data.frame` now allows the `f` argument to be specified as a formula.
- `sprintf` now warns on arguments unused by the format string.
- New palettes `"Rocket"` and `"Mako"` for `hcl.colors()` (approximating palettes of the same name from the **viridisLite** package).
  Contributed by Achim Zeileis.
- The base environment and its namespace are now locked (so one can no longer add bindings to these or remove from these).
- `Rterm` handling of multi-byte characters has been improved, allowing use of such characters when supported by the current locale.
- `Rterm` now accepts `ALT+ +xxxxxxxx` sequences to enter Unicode characters as hex digits.
- Environment variable `LC_ALL` on Windows now takes precedence over `LC_CTYPE` and variables for other supported categories, matching the POSIX behaviour.
- `duplicated()` and `anyDuplicated()` are now optimized for integer and real vectors that are known to be sorted via the ALTREP framework. Contributed by Gabriel Becker via PR#17993.

Graphics:

- The graphics engine version, `R_GE_version`, has been bumped to `14` and so packages that provide graphics devices should be reinstalled.

- Graphics devices should now specify `deviceVersion` to indicate what version of the graphics engine they support.
- Graphics devices can now specify `deviceClip`. If `TRUE`, the graphics engine will never perform any clipping of output itself.
  The clipping that the graphics engine does perform (for both `canClip = TRUE` and `canClip = FALSE`) has been improved to avoid producing unnecessary artifacts in clipped output.
- The **grid** package now allows `gpar(fill)` to be a `linearGradient()`, a `radialGradient()`, or a `pattern()`. The `viewport(clip)` can now also be a grob, which defines a clipping path, and there is a new `viewport(mask)` that can also be a grob, which defines a mask.
  These new features are only supported so far on the Cairo-based graphics devices and on the `pdf()` device.
- (Not Windows.) A warning is given when a Cairo-based type is specified for a `png()`, `jpeg()`, `tiff()` or `bmp()` device but Cairo is unsupported (so `type = "Xlib"` is tried instead).
- `grSoftVersion()` now reports the versions of FreeType and FontConfig if they are used directly (not *via* Pango), as is most commonly done on macOS.

C-Level Facilities:

- The *standalone* '`libRmath`' math library and R's C API now provide `log1pexp()` again as documented, and gain `log1mexp()`.

Installation on a Unix-Alike:

- `configure` checks for a program `pkgconf` if program `pkg-config` is not found. These are now only looked for on the path (like almost all other programs) so if needed specify a full path to the command in `PKG_CONFIG`, for example in file '`config.site`'.
- C99 function `iswblank` is required – it was last seen missing ca 2003 so the workaround has been removed.
- There are new `configure` options '`--with-internal-iswxxxxx`', '`--with-internal-towlower`' and '`--with-internal-wcwidth`' which allows the system functions for wide-character classification, case-switching and width (`wcwidth` and `wcswidth`) to be replaced by internal ones. The first has long been used on macOS, AIX (and Windows) but this enables it to be unselected there and selected for other platforms (it is the new default on Solaris). The second is new in this version of R and is selected by default on macOS and Solaris. The third has long been the default and remains so as it contains customizations for East Asian languages.
  System versions of these functions are often minimally implemented (sometimes only for ASCII characters) and may not cover the full range of Unicode points: for example Solaris (and Windows) only cover the Basic Multilingual Plane.
- Cairo installations without X11 are more likely to be detected by `configure`, when the file-based Cairo graphics devices will be available but not `X11(type = "cairo")`.
- There is a new `configure` option '`--with-static-cairo`' which is the default on macOS. This should be used when only static cairo (and where relevant, Pango) libraries are available.
- Cairo-based graphics devices on platforms without Pango but with FreeType/FontConfig will make use of the latter for font selection.

Link-Time Optimization on a Unix-Alike:

- Configuring with flag '`--enable-lto=R`' now also uses LTO when installing the recommended packages.

- `R CMD INSTALL` and `R CMD SHLIB` have a new flag '`--use-LTO`' to use LTO when compiling code, for use with R configured with '`--enable-lto=R`'. For R configured with '`--enable-lto`', they have the new flag '`--no-use-LTO`'.
  Packages can opt in or out of LTO compilation *via* a '`UseLTO`' field in the '`DESCRIPTION`' file. (As usual this can be overridden by the command-line flags.)

Building R on Windows:

- for GCC $\geq$ 8, `FC_LEN_T` is defined in '`config.h`' and hence character lengths are passed from C to Fortran in *inter alia* BLAS and LAPACK calls.
- There is a new text file '`src/gnuwin32/README.compilation`', which outlines how C/Fortran code compilation is organized and documents new features:
  - R can be built with Link-Time Optimization with a suitable compiler – doing so with GCC 9.2 showed several inconsistencies which have been corrected.
  - There is support for cross-compiling the C and Fortran code in R and standard packages on suitable (Linux) platforms. This is mainly intended to allow developers to test later versions of compilers – for example using GCC 9.2 or 10.x has detected issues that GCC 8.3 in Rtools40 does not.
  - There is experimental support for cross-building R packages with C, C++ and/or Fortran code.
- The R installer can now be optionally built to support a single architecture (only 64-bit or only 32-bit).

Package Installation:

- The default C++ standard has been changed to C++14 where available (which it is on all currently checked platforms): if not (as before) C++11 is used if available otherwise C++ is not supported.
  Packages which specify C++11 will still be installed using C++11.
  C++14 compilers may give deprecation warnings, most often for `std::random_shuffle` (deprecated in C++14 and removed in C++17). Either specify C++11 (see 'Writing R Extensions') or modernize the code and if needed specify C++14. The latter has been supported since R 3.4.0 so the package's '`DESCRIPTION`' would need to include something like

  ```
  Depends: R (>= 3.4)
  ```

Package Installation on Windows:

- `R CMD INSTALL` and `R CMD SHLIB` make use of their flag '`--use-LTO`' when the '`LTO_OPT`' make macro is set in file '`etc/${R_ARCH}/Makeconf`' or in a personal/site '`Makevars`' file. (For details see 'Writing R Extensions' §4.5.)
  This provides a valuable check on code consistency. It does work with GCC 8.3 as in Rtools40, but that does not detect everything the CRAN checks with current GCC do.

Package Installation on macOS:

- The default personal library directory on builds with '`--enable-aqua`' (including CRAN builds) now differs by CPU type, one of

  ```
  ~/Library/R/x86_64/x.y/library
  ~/Library/R/arm64/x.y/library
  ```

  This uses the CPU type R (and hence the packages) were built for, so when a '`x86_64`' build of R is run under Rosetta emulation on an '`arm64`' Mac, the first is used.

Utilities:

- `R CMD check` can now scan package functions for bogus `return` statements, which were possibly intended as `return()` calls (wish of PR#17180, patch by Sebastian Meyer). This check can be activated via the new environment variable `_R_CHECK_BOGUS_RETURN_`, true for `--as-cran`.
- `R CMD build` omits tarballs and binaries of previous builds from the top-level package directory. (PR#17828, patch by Sebastian Meyer.)
- `R CMD check` now runs sanity checks on the use of '`LazyData`', for example that a '`data`' directory is present and that '`LazyDataCompression`' is not specified without '`LazyData`' and has a documented value. For packages with large LazyData databases without specifying '`LazyDataCompression`', there is a reference to the code given in 'Writing R Extensions' §1.1.6 to test the choice of compression (as in all the CRAN packages tested a non-default method was preferred).
- `R CMD build` removes '`LazyData`' and '`LazyDataCompression`' fields from the '`DESCRIPTION`' file of packages without a '`data`' directory.

Encoding-Related Changes:

- The parser now treats '`\Unnnnnnnn`' escapes larger than the upper limit for Unicode points ('`\U10FFFF`') as an error as they cannot be represented by valid UTF-8.
  Where such escapes are used for outputting non-printable (including unassigned) characters, 6 hex digits are used (rather than 8 with leading zeros). For clarity, braces are used, for example '`\U{0effff}`'.
- The parser now looks for non-ASCII spaces on Solaris (as previously on most other OSes).
- There are warnings (including from the parser) on the use of unpaired surrogate Unicode points such as '`\uD834`'. (These cannot be converted to valid UTF-8.)
- Functions `nchar()`, `tolower()`, `toupper()` and `chartr()` and those using regular expressions have more support for inputs with a marked Latin-1 encoding.
- The character-classification functions used (by default) to replace the system `iswxxxxx` functions on Windows, macOS and AIX have been updated to Unicode 13.0.0.
  The character-width tables have been updated to include new assignments in Unicode 13.0.0. This included treating all control characters as having zero width.
- The code for evaluating default (extended) regular expressions now uses the same character-classification functions as the rest of R (previously they differed on Windows, macOS and AIX).
- There is a build-time option to replace the system's wide-character `wctrans` C function by tables shipped with R: use `configure` option '`--with-internal-towlower`' or (on Windows) '`-DUSE_RI18N_CASE`' in '`CFLAGS`' when building R. This may be needed to allow `tolower()` and `toupper()` to work with Unicode characters beyond the Basic Multilingual Plane where not supported by system functions (e.g. on Solaris where it is the new default).
- R is more careful when truncating UTF-8 and other multi-byte strings that are too long to be printed, passed to the system or libraries or placed into an internal buffer. Truncation will no longer produce incomplete multibyte characters.

Deprecated and Defunct:

- Function `plclust()` from the package **stats** and `package.dependencies()`, `pkgDepends()`, `getDepList()`, `installFoundDepends()`, and `vignetteDepends()` from package **tools** are defunct.
- Defunct functions `checkNEWS()` and `readNEWS()` from package **tools** and `CRAN.packages()` from **utils** have been removed.

- `R CMD config CXXCPP` is defunct (it was deprecated in R 3.6.2).
- `parallel::detectCores()` drops support for IRIX (retired in 2013).
- The `LINPACK` argument to `chol.default()`, `chol2inv()`, `solve.default()` and `svd()` has been defunct since R 3.1.0. It was silently ignored up to R 4.0.3 but now gives an error.
- Subsetting/indexing, such as `ddd[*]` or `ddd$x` on a `DOTSXP` (dot-dot-dot) object `ddd` has been disabled; it worked by accident only and was undocumented.

Bug Fixes:

- Many more C-level allocations (mainly by `malloc` and `strdup`) are checked for success with suitable alternative actions.
- Bug fix for `replayPlot()`; this was turning off graphics engine display list recording if a recorded plot was replayed in the same session. The impact of the bug became visible if resize the device after replay OR if attempted another `savePlot()` after replay (empty display list means empty screen on resize or empty saved plot).
- `R CMD check` etc now warn when a package exports non-existing S4 classes or methods, also in case of no "methods" presence. (Reported by Alex Bertram; reproducible example and patch by Sebastian Meyer in PR#16662.)
- `boxplot()` now also accepts `call`s for labels such as `ylab`, the same as `plot()`. (Reported by Marius Hofert.)
- The help page for `xtabs()` now correctly states that `addNA` is setting `na.action = na.pass` among others. (Reported as PR#17770 by Thomas Soeiro.)
- The `R CMD check <pkg>` gives a longer and more comprehensible message when '`DESCRIPTION`' misses dependencies, e.g., in `Imports:`. (Thanks to the contributors of PR#17179.)
- `update.default()` now calls the generic `update()` on the formula to work correctly for models with extended formulas. (As reported and suggested by Neal Fultz in PR#17865.)
- The horizontal position of leaves in a dendrogram is now correct also with `center = FALSE`. (PR#14938, patch from Sebastian Meyer.)
- `all.equal.POSIXt()` no longer warns about and subsequently ignores inconsistent `"tzone"` attributes, but describes the difference in its return value (PR#17277). This check can be disabled *via* the new argument `check.tzone = FALSE` as suggested by Sebastian Meyer.
- `as.POSIXct()` now populates the `"tzone"` attribute from its `tz` argument when `x` is a logical vector consisting entirely of `NA` values.
- `x[[2^31]] <- v` now works. (Thanks to the report and patch by Suharto Anggono in PR#17330.)
- In log-scale graphics, `axis()` ticks and label positions are now computed more carefully and symmetrically in their range, typically providing *more* ticks, fulfilling wishes in PR#17936. The change really corresponds to an improved `axisTicks()` (package **grDevices**), potentially influencing **grid** and **lattice**, for example.
- `qnorm(<very large negative>, log.p=TRUE)` is now correct to at least five digits where it was catastrophically wrong, previously.
- `sum(df)` and similar `"Summary"`- and `"Math"`-group member functions now work for data frames `df` with `logical` columns, notably also of zero rows. (Reported to R-devel by Martin "b706".)
- `unsplit()` had trouble with tibbles due to unsound use of `rep(NA, len)`-indexing, which should use `NA_integer_` (Reported to R-devel by Mario Annau.)
- `pnorm(x, log.p = TRUE)` underflows to `-Inf` slightly later.

- `show(<hidden S4 generic>)` prints better and without quotes for non-hidden S4 generics.
- `read.table()` and relatives treated an "NA" column name as missing when `check.names = FALSE` PR#18007.
- Parsing strings containing UTF-16 surrogate pairs such as `"\uD834\uDD1E"` works better on some (uncommon) platforms. `sprintf("%X", utf8ToInt("\uD834\uDD1E"))` should now give `"1D11E"` on all platforms.
- `identical(x,y)` is no longer true for differing `DOTSXP` objects, fixing PR#18032.
- `str()` now works correctly for `DOTSXP` and related exotics, even when these are doomed.
  Additionally, it no longer fails for `lists` with a `class` and "irregular" method definitions such that e.g. `lapply(*)` will necessarily fail, as currently for different **igraph** objects.
- Message translation domains, e.g., for errors and warnings, are now correctly determined also when e.g., a **base** function is called from "top-level" function (i.e., defined in `globalenv()`), thanks to a patch from Joris Goosen fixing PR#17998.
- Too long lines in environment files (e.g., '`Renviron`') no longer crash R. This limit has been increased to 100,000 bytes. (PR#18001.)
- There is a further workaround for FreeType giving incorrect italic font faces with cairo-based graphics devices on macOS.
- `add_datalist(*, force = TRUE)` (from package **tools**) now actually updates an existing '`data/datalist`' file for new content. (Thanks to a report and patch by Sebastian Meyer in PR#18048.)
- `cut.Date()` and `cut.POSIXt()` could produce an empty last interval for `breaks = "months"` or `breaks = "years"`. (Reported as PR#18053 by Christopher Carbone.)
- Detection of the encoding of 'regular' macOS locales such as '`en_US`' (which is UTF-8) had been broken by a macOS change: fortunately these are now rarely used with '`en_US.UTF-8`' being preferred.
- `sub()` and `gsub(pattern, repl, x, *)` now keep attributes of `x` such as `names()` also when `pattern` is `NA` (PR#18079).
- Time differences (`"difftime"` objects) get a replacement and a `rep()` method to keep `"units"` consistent. (Thanks to a report and patch by Nicolas Bennett in PR#18066.)
- The `\RdOpts` macro, setting defaults for `\Sexpr` options in an Rd file, had been ineffective since R 2.12.0: it now works again. (Thanks to a report and patch by Sebastian Meyer in PR#18073.)
- `mclapply` and `pvec` no longer accidentally terminate parallel processes started before by `mcparallel` or related calls in package **parallel** (PR#18078).
- `grep` and other functions for evaluating (extended) regular expressions handle in Unicode also strings not explicitly flagged UTF-8, but flagged native when running in UTF-8 locale.
- Fixed a crash in `fifo` implementation on Windows (PR#18031).
- Binary mode in `fifo` on Windows is now properly detected from argument `open` (PR#15600, PR#18031).



Changes in R 4.0.5

Bug Fixes:

- The change to the internal table in R 4.0.4 for `iswprint` has been reverted: it contained some errors in printability of 'East Asian' characters.

- For packages using 'LazyData', R CMD build ignored the '--resave-data' option and the 'BuildResaveData' field of the 'DESCRIPTION' file (in R versions 4.0.0 to 4.0.4).



Changes in R 4.0.4

New Features:

- File 'share/texmf/tex/latex/jss.cls' has been updated to work with LaTeX versions since Oct 2020.
- Unicode character width tables (as used by nchar(, type = "w")) have been updated to Unicode 12.1 by Brodie Gaslam (PR#17781), including many emoji.
- The internal table for iswprint (used on Windows, macOS and AIX) has been updated to include many recent Unicode characters.

Installation on a Unix-Alike:

- If an external BLAS is specified by '--with-blas=foo' or *via* environment variable BLAS_LIBS is not found, this is now a configuration error. The previous behaviour was not clear from the documentation: it was to continue the search as if '--with-blas=yes' was specified.

Bug Fixes:

- all.equal(x,y) now "sees" the two different NAs in factors, thanks to Bill Dunlap and others in PR#17897.
- (~ NULL)[1] and similar formula subsetting now works, thanks to a report and patch by Henrik Bengtsson in PR#17935. Additionally, subsetting leaving an empty formula now works too, thanks to suggestions by Suharto Anggono.
- .traceback(n) keeps source references again, as before R 4.0.0, fixing a regression; introduced by the PR#17580, reported including two patch proposals by Brodie Gaslam.
- unlist(plst, recursive=FALSE) no longer drops content for pairlists with list components, thanks to the report and patch by Suharto Anggono in PR#17950.
- iconvlist() now also works on MUSL based (Linux) systems, from a report and patch suggestion by Wesley Chan in PR#17970.
- round() and signif() no longer tolerate wrong argument names, notably in 1-argument calls; reported by Shane Mueller on R-devel (mailing list); later reported as PR#17976.
- .Machine has longdouble.* elements only if capabilities("long.double") is true, as documented. (Previously they were included if the platform had long double identical to double, as ARM does.)
- p.adjust(numeric(), n=0) now works, fixing PR#18002.
- identical(x,y) no longer prints "Unknown Type .." for typeof(x) == "..." objects.
- Fix (auto-)print()ing of named complex vectors, see PR#17868 and PR#18019.
- all.equal(<language>, <...>) now works, fixing PR#18029.
- as.data.frame.list(L, row.names=NULL) now behaves in line with data.frame(), disregarding names of components of L, fixing PR#18034, reported by Kevin Tappe.
- checkRdaFiles(ff)$version is now correct also when ff contains files of different versions, thanks to a report and patch from Sebastian Meyer in PR#18041.
- macOS: Quartz device live drawing could fail (no plot is shown) if the system changes the drawing context after view update (often the case since macOS Big Sur). System log may show "CGContextDelegateCreateForContext: invalid context" error.



Changes in R 4.0.3

New Features:

- On platforms using `configure` option '`--with-internal-tzcode`', additional values `"internal"` and (on macOS only) `"macOS"` are accepted for the environment variable `TZDIR`. (See `?TZDIR`.)
  On macOS, `"macOS"` is used by default if the system timezone database is a newer version than that in the R installation.
- When `install.packages(type = "source")` fails to find a package in a repository it mentions package versions which are excluded by their R version requirement and links to hints on why a package might not be found.
- The default value for `options("timeout")` can be set from environment variable `R_DEFAULT_INTERNET_TIMEOUT`, still defaulting to 60 (seconds) if that is not set or invalid.
  This may be needed when child R processes are doing downloads, for example during the installation of source packages which download jars or other forms of data.

Link-Time Optimization on a Unix-Alike:

- There is now support for parallelized Link-Time Optimization (LTO) with GCC and for 'thin' LTO with `clang` *via* setting the '`LTO`' macro.
- There is support for setting a different LTO flag for the Fortran compiler, including to empty when mixing `clang` and `gfortran` (as on macOS). See file '`config.site`'.
- There is a new '`LTO_LD`' macro to set linker options for LTO compilation, for example to select an alternative linker or to parallelize thin LTO.

Deprecated and Defunct:

- The `LINPACK` argument to `chol.default()`, `chol2inv()`, `solve.default()` and `svd()` has been defunct since R 3.1.0. Using it now gives a warning which will become an error in R 4.1.0.

Bug Fixes:

- The code mitigating stack overflow with PCRE regexps on very long strings is enabled for PCRE2 < 10.30 also when JIT is enabled, since stack overflows have been seen in that case.
- Fix to correctly show the group labels in `dotchart()` (which where lost in the `ylab` improvement for R 4.0.0).
- `addmargins(*, ..)` now also works when `fn()` is a local function, thanks to bug report and patch PR#17124 from Alex Bertram.
- `rank(x)` and hence `sort(x)` now work when `x` is an object (as per `is.object(x)`) of type `"raw"` *and* provides a valid `` `[` `` method, e.g., for `gmp::as.bigz(.)` numbers.
- `chisq.test(*, simulate.p.value=TRUE)` and `r2dtable()` now work correctly for large table entries (in the millions). Reported by Sebastian Meyer and investigated by more helpers in PR#16184.
- Low-level socket read/write operations have been fixed to correctly signal communication errors. Previously, such errors could lead to a segfault due to invalid memory access. Reported and debugged by Dmitriy Selivanov in PR#17850.
- `quantile(x, pr)` works more consistently for `pr` values slightly outside [0,1], thanks to Suharto Anggono's PR#17891.
  Further, `quantile(x, prN, names=FALSE)` now works even when `prN` contains `NA`s, thanks to Anggono's PR#17892. Ditto for ordered factors or `Date` objects when `type = 1` or `3`, thanks to PR#17899.

- Internet access based on libcurl, including `curlGetHeaders()`, was not respecting the `"timeout"` option. If this causes unanticipated timeouts, consider increasing the default by setting `R_DEFAULT_INTERNET_TIMEOUT`.
- `as.Date(<char>)` now also works with an initial `""`, thanks to Michael Chirico's PR#17909.
- `isS3stdGeneric(f)` now detects an S3 generic also when it is `trace()`d, thanks to Gabe Becker's PR#17917.
- `R_allocLD()` has been fixed to return memory aligned for long double type PR#16534.
- `fisher.test()` no longer segfaults when called again after its internal stack has been exceeded PR#17904.
- Accessing a long vector represented by a compact integer sequence no longer segfaults (reported and debugged by Hugh Parsonage).
- `duplicated()` now works also for strings with multiple encodings inside a single vector PR#17809.
- `phyper(11, 15, 0, 12, log.p=TRUE)` no longer gives `NaN`; reported as PR#17271 by Alexey Stukalov.
- Fix incorrect calculation in `logLik.nls()` PR#16100, patch from Sebastian Meyer.
- A very old bug could cause a segfault in `model.matrix()` when terms involved logical variables. Part of PR#17879.
- `model.frame.default()` allowed `data = 1`, leading to involuntary variable capture (rest of PR#17879).
- `tar()` no longer skips non-directory files, thanks to a patch by Sebastian Meyer, fixing the remaining part of PR#16716.



Changes in R 4.0.2

Utilities:

- `R CMD check` skips vignette re-building (with a warning) if the '`VignetteBuilder`' package(s) are not available.

Bug Fixes:

- Paths with non-ASCII characters caused problems for package loading on Windows PR#17833.
- Using **tcltk** widgets no longer crashes R on Windows.
- `source(*, echo=TRUE)` no longer fails in some cases with empty lines; reported by Bill Dunlap in PR#17769.
- `on.exit()` now correctly matches named arguments, thanks to PR#17815 (including patch) by Brodie Gaslam.
- `regexpr(*, perl=TRUE)` no longer returns incorrect positions into text containing characters outside of the Unicode Basic Multilingual Plane on Windows.



Changes in R 4.0.1

New Features:

- `paste()` and `paste0()` gain a new optional argument `recycle0`. When set to true, zero-length arguments are recycled leading to `character(0)` after the `sep`-concatenation, i.e., to the empty string `""` if `collapse` is a string and to the zero-length value `character(0)` when `collapse = NULL`.
  A package whose code uses this should depend on '`R (>= 4.0.1)`'.
- The `summary(<warnings>)` method now maps the counts correctly to the warning messages.

Bug Fixes:

- `aov(frml, ...)` now also works where the `formula` deparses to more than 500 characters, thanks to a report and patch proposal by Jan Hauffa.
- Fix a dozen places (code, examples) as `Sys.setlocale()` returns the new rather than the previous setting.
- Fix for adding two complex **grid** units via `sum()`. Thanks to Gu Zuguang for the report and Thomas Lin Pedersen for the patch.
- Fix `parallel::mclapply(..., mc.preschedule=FALSE)` to handle raw vector results correctly. PR#17779
- Computing the `base` value, i.e., 2, "everywhere", now uses `FLT_RADIX`, as the original '`machar`' code looped indefinitely on the ppc64 architecture for the `longdouble` case.
- In R 4.0.0, `sort.list(x)` when `is.object(x)` was true, e.g., for `x <- I(letters)`, was accidentally using `method = "radix"`. Consequently, e.g., `merge(<data.frame>)` was much slower than previously; reported in PR#17794.
- `plot(y ~ x, ylab = quote(y[i]))` now works, as e.g., for `xlab`; related to PR#10525.
- `parallel::detect.cores(all.tests = TRUE)` tries a matching OS name before the other tests (which were intended only for unknown OSes).
- Parse data for raw strings is now recorded correctly. Reported by Gabor Csardi.



Changes in R 4.0.0

Significant User-Visible Changes:

- Packages need to be (re-)installed under this version (4.0.0) of R.
- `matrix` objects now also inherit from class `"array"`, so e.g., `class(diag(1))` is `c("matrix","array")`. This invalidates code incorrectly assuming that `class(matrix_obj))` has length one.
  S3 methods for class `"array"` are now dispatched for `matrix` objects.
- There is a new syntax for specifying *raw* character constants similar to the one used in C++: `r"(...)"` with `...` any character sequence not containing the sequence '`)"`'. This makes it easier to write strings that contain backslashes or both single and double quotes. For more details see `?Quotes`.
- R now uses a '`stringsAsFactors = FALSE`' default, and hence by default no longer converts strings to factors in calls to `data.frame()` and `read.table()`.
  A large number of packages relied on the previous behaviour and so have needed/will need updating.
- The `plot()` S3 generic function is now in package **base** rather than package **graphics**, as it is reasonable to have methods that do not use the **graphics** package. The generic is currently re-exported from the **graphics** namespace to allow packages importing it from there to continue working, but this may change in future.

Packages which define S4 generics for `plot()` should be re-installed and package code using such generics from other packages needs to ensure that they are imported rather than rely on their being looked for on the search path (as in a namespace, the base namespace has precedence over the search path).

Reference Counting:

- Reference counting is now used instead of the `NAMED` mechanism for determining when objects can be safely mutated in base C code. This reduces the need for copying in some cases and should allow further optimizations in the future. It should help make the internal code easier to maintain.
  This change is expected to have almost no impact on packages using supported coding practices in their C/C++ code.

Migration to PCRE2:

- This version of R is built against the PCRE2 library for Perl-like regular expressions, if available. (On non-Windows platforms PCRE1 can optionally be used if PCRE2 is not available at build time.) The version of PCRE in use can be obtained *via* `extSoftVersion()`: PCRE1 (formerly known as 'PCRE') has versions <= 8, PCRE2 versions >= 10.
- Making PCRE2 available when building R from source is strongly recommended (preferably version 10.30 or later) as PCRE1 is no longer developed: version 8.44 is 'likely to be the final release'.
- PCRE2 reports errors for some regular expressions that were accepted by PCRE1. A hyphen now has to be escaped in a character class to be interpreted as a literal (unless first or last in the class definition). '`\R`', '`\B`' and '`\X`' are no longer allowed in character classes (PCRE1 treated these as literals).
- Option `PCRE_study` is no longer used with PCRE2, and is reported as `FALSE` when that is in use.

New Features:

- `assertError()` and `assertWarning()` (in package **tools**) can now check for *specific* error or warning classes *via* the new optional second argument `classes` (which is not back compatible with previous use of an unnamed second argument).
- `DF2formula()`, the utility for the data frame method of `formula()`, now works without parsing and explicit evaluation, starting from Suharto Anggono's suggestion in PR#17555.
- `approxfun()` and `approx()` gain a new argument `na.rm` defaulting to true. If set to false, missing `y` values now propagate into the interpolated values.
- Long vectors are now supported as the `seq` argument of a `for()` loop.
- `str(x)` gets a new `deparse.lines` option with a default to speed it up when `x` is a large `call` object.
- The internal traceback object produced when an error is signalled (`.Traceback`), now contains the `call`s rather than the `deparse()`*d* calls, deferring the deparsing to the user-level functions `.traceback()` and `traceback()`. This fulfils the wish of PR#17580, reported including two patch proposals by Brodie Gaslam.
- `data.matrix()` now converts character columns to factors and from this to integers.
- `package.skeleton()` now explicitly lists all exports in the '`NAMESPACE`' file.
- New function `.S3method()` to register S3 methods in R scripts.
- `file.path()` has some support for file paths not in the session encoding, e.g. with UTF-8 inputs in a non-UTF-8 locale the output is marked as UTF-8.

- Most functions with file-path inputs will give an explicit error if a file-path input in a marked encoding cannot be translated (to the native encoding or in some cases on Windows to UTF-8), rather than translate to a different file path using escapes. Some (such as `dir.exists()`, `file.exists()`, `file.access()`, `file.info()`, `list.files()`, `normalizePath()` and `path.expand()`) treat this like any other non-existent file, often with a warning.
- There is a new help document accessed by `help("file path encoding")` detailing how file paths with marked encodings are handled.
- New function `list2DF()` for creating data frames from lists of variables.
- `iconv()` has a new option `sub = "Unicode"` to translate UTF-8 input invalid in the '`to`' encoding using '`<U+xxxx>`' escapes.
- There is a new function `infoRDS()` providing information about the serialization format of a serialized object.
- S3 method lookup now by default skips the elements of the search path between the global and base environments.
- Added an argument `add_datalist(*, small.size = 0)` to allow the creation of a '`data/datalist`' file even when the total size of the data sets is small.
- The backquote function `bquote()` has a new argument `splice` to enable splicing a computed list of values into an expression, like `,@` in LISP's backquote.
- The formula interface to `t.test()` and `wilcox.test()` has been extended to handle one-sample and paired tests.
- The `palette()` function has a new default set of colours (which are less saturated and have better accessibility properties). There are also some new built-in palettes, which are listed by the new `palette.pals()` function. These include the old default palette under the name `"R3"`. Finally, the new `palette.colors()` function allows a subset of colours to be selected from any of the built-in palettes.
- `n2mfrow()` gains an option `asp = 1` to specify the aspect ratio, fulfilling the wish and extending the proposal of Michael Chirico in PR#17648.
- For `head(x, n)` and `tail()` the default and other S3 methods notably for *vector* `n`, e.g. to get a "corner" of a matrix, has been extended to `array`'s of higher dimension thanks to the patch proposal by Gabe Becker in PR#17652. Consequently, optional argument `addrownums` is deprecated and replaced by the (more general) argument `keepnums`. An invalid second argument `n` now leads to typically more easily readable error messages.
- New function `.class2()` provides the full character vector of class names used for S3 method dispatch.
- Printing `methods(..)` now uses a new `format()` method.
- `sort.list(x)` now works for non-atomic objects `x` and `method = "auto"` (the default) or `"radix"` in cases `order(x)` works, typically via a `xtfrm()` method.
- Where they are available, `writeBin()` allows long vectors.
- New function `deparse1()` produces one string, wrapping `deparse()`, to be used typically in `deparse1(substitute(*))`, e.g., to fix PR#17671.
- `wilcox.test()` enhancements: In the (non-paired) two-sample case, `Inf` values are treated as very large for robustness consistency. If exact computations are used, the result now has `"exact"` in the `method` element of its return value. New arguments `tol.root` and `digits.rank` where the latter may be used for stability to treat very close numbers as ties.
- `readBin()` and `writeBin()` now report an error for an invalid `endian` value. The affected code needs to be fixed with care as the old undocumented behavior was to swap endianness in such cases.

- `sequence()` is now an S3 generic with an internally implemented default method, and gains arguments to generate more complex sequences. Based on code from the **S4Vectors** Bioconductor package and the advice of Hervé Pagès.
- `print()`'s default method and many other methods (by calling the default eventually and passing `...`) now make use of a new optional `width` argument, avoiding the need for the user to set and reset `options("width")`.
- `memDecompress()` supports the RFC 1952 format (e.g. in-memory copies of `gzip`-compressed files) as well as RFC 1950.
- `memCompress()` and `memDecompress()` support long raw vectors for types `"gzip"` and `"zx"`.
- `sweep()` and `slice.index()` can now use names of dimnames for their `MARGIN` argument (`apply` has had this for almost a decade).
- New function `proportions()` and `marginSums()`. These should replace the unfortunately named `prop.table()` and `margin.table()`. They are drop-in replacements, but also add named-margin functionality. The old function names are retained as aliases for back-compatibility.
- Functions `rbinom()`, `rgeom()`, `rhyper()`, `rpois()`, `rnbinom()`, `rsignrank()` and `rwilcox()` which have returned integer since R 3.0.0 and hence `NA` when the numbers would have been outside the integer range, now return double vectors (without NAs, typically) in these cases.
- `matplot(x,y)` (and hence `matlines()` and `matpoints()`) now call the corresponding methods of `plot()` and `lines()`, e.g, when `x` is a `"Date"` or `"POSIXct"` object; prompted by Spencer Graves' suggestion.
- `stopifnot()` now allows customizing error messages via argument names, thanks to a patch proposal by Neal Fultz in PR#17688.
- `unlink()` gains a new argument `expand` to disable wildcard and tilde expansion. Elements of `x` of value `"~"` are now ignored.
- `mle()` in the **stats4** package has had its interface extended so that arguments to the negative log-likelihood function can be one or more vectors, with similar conventions applying to bounds, start values, and parameter values to be kept fixed. This required a minor extension to class `"mle"`, so saved objects from earlier versions may need to be recomputed.
- The default for `pdf()` is now `useDingbats = FALSE`.
- The default fill colour for `hist()` and `boxplot()` is now `col = "lightgray"`.
- The default order of the levels on the y-axis for `spineplot()` and `cdplot()` has been reversed.
- If the `R_ALWAYS_INSTALL_TESTS` environment variable is set to a true value, `R CMD INSTALL` behaves as if the '`--install-tests`' option is always specified. Thanks to Reinhold Koch for the suggestion.
- New function `R_user_dir()` in package **tools** suggests paths appropriate for storing R-related user-specific data, configuration and cache files.
- `capabilities()` gains a new logical option `Xchk` to avoid warnings about X11-related capabilities.
- The internal implementation of **grid** units has changed, but the only visible effects at user-level should be
  - a slightly different print format for some units (especially unit arithmetic),
  - faster performance (for unit operations) and
  - two new functions `unitType()` and `unit.psum()`.

  Based on code contributed by Thomas Lin Pedersen.

- When internal dispatch for `rep.int()` and `rep_len()` fails, there is an attempt to dispatch on the equivalent call to `rep()`.
- Object `.Machine` now contains new `longdouble.*` entries (when R uses long doubles internally).
- `news()` has been enhanced to cover the news on R 3.x and 2.x.
- For consistency, `N <- NULL; N[[1]] <- val` now turns `N` into a `list` also when `val)` has length one. This enables `dimnames(r1)[[1]] <- "R1"` for a 1-row matrix `r1`, fixing PR#17719 reported by Serguei Sokol.
- `deparse(..)`, `dump(..)`, and `dput(x, control = "all")` now include control option `"digits17"` which typically ensures 1:1 invertibility. New option `control = "exact"` ensures numeric exact invertibility via `"hexNumeric"`.
- When loading data sets via `read.table()`, `data()` now uses '`LC_COLLATE=C`' to ensure locale-independent results for possible string-to-factor conversions.
- A server socket connection, a new connection type representing a listening server socket, is created via `serverSocket()` and can accept multiple socket connections via `socketAccept()`.
- New function `socketTimeout()` changes the connection timeout of a socket connection.
- The time needed to start a homogeneous '`PSOCK`' cluster on '`localhost`' with many nodes has been significantly reduced (package **parallel**).
- New `globalCallingHandlers()` function to establish global condition handlers. This allows registering default handlers for specific condition classes. Developed in collaboration with Lionel Henry.
- New function `tryInvokeRestart()` to invoke a specified restart if one is available and return without signaling an error if no such restart is found. Contributed by Lionel Henry in PR#17598.
- `str(x)` now shows the length of `attributes` in some cases for a data frame `x`.
- `Rprof()` gains a new argument `filter.callframes` to request that intervening call frames due to lazy evaluation or explicit `eval()` calls be omitted from the recorded profile data. Contributed by Lionel Henry in PR#17595.
- The handling of `${FOO-bar}` and `${FOO:-bar}` in '`Renviron`' files now follows POSIX shells (at least on a Unix-alike), so the first treats empty environment variables as set and the second does not. Previously both ignored empty variables. There are several uses of the first form in '`etc/Renviron`'.
- New `classes` argument for `suppressWarnings()` and `suppressMessages()` to selectively suppress only warnings or messages that inherit from particular classes. Based on patch from Lionel Henry submitted with PR#17619.
- New function `activeBindingFunction()` retrieves the function of an active binding.
- New `"cairoFT"` and `"pango"` components in the output of `grSoftVersion()`.
- New argument `symbolfamily` in cairo-based graphics devices and new function `cairoSymbolFont()` that can be used to provide the value for that argument.

Windows:

- `Rterm` now works also when invoked from MSYS2 terminals. Line editing is possible when command `winpty` is installed.
- `normalizePath()` now resolves symbolic links and normalizes case of long names of path elements in case-insensitive folders (PR#17165).
- `md5sum()` supports UTF-8 file names with characters that cannot be translated to the native encoding (PR#17633).

- `Rterm` gains a new option '`--workspace`' to specify the workspace to be restored. This allows equals to be part of the name when opening *via* Windows file associations (reported by Christian Asseburg).
- `Rterm` now accepts `ALT+xxx` sequences also with NumLock on. Tilde can be pasted with an Italian keyboard (PR#17679).
- R falls back to copying when junction creation fails during package checking (patch from Duncan Murdoch).

Deprecated and Defunct:

- Make macro '`F77_VISIBILITY`' has been removed and replaced by '`F_VISIBILITY`'.
- Make macros '`F77`', '`FCPIFCPLAGS`' and '`SHLIB_OPENMP_FCFLAGS`' have been removed and replaced by '`FC`', '`FPICFLAGS`' and '`SHLIB_OPENMP_FFLAGS`' respectively. (Most `make` programs will set '`F77`' to the value of '`FC`', which is set for package compilation. But portable code should not rely on this.)
- The deprecated support for specifying C++98 for package installation has been removed.
- `R CMD config` no longer knows about the unused settings '`F77`' and '`FCPIFCPLAGS`', nor '`CXX98`' and similar.
- Either PCRE2 or PCRE1 >= 8.32 (Nov 2012) is required: the deprecated provision for 8.20–8.31 has been removed.
- Defunct functions `mem.limits()`, `.readRDS()`, `.saveRDS()`, `.find.package()`, and `.path.package()` from package **base** and `allGenerics()`, `getAccess()`, `getAllMethods()`, `getClassName()`, `getClassPackage()`, `getExtends()`, `getProperties()`, `getPrototype()`, `getSubclasses()`, `getVirtual()`, `mlistMetaName()`, `removeMethodsObject()`, `seemsS4Object()`, `traceOff()`, and `traceOn()` from **methods** have been removed.

C-Level Facilities:

- `installChar` is now remapped in '`Rinternals.h`' to `installTrChar`, of which it has been a wrapper since R 3.6.0. Neither are part of the API, but packages using `installChar` can replace it if they depend on '`R >= 3.6.2`'.
- Header '`R_ext/Print.h`' defines '`R_USE_C99_IN_CXX`' and hence exposes `Rvprintf` and `REvprintf` if used with a C++11 (or later) compiler.
- There are new Fortran subroutines `dblepr1`, `realpr1` and `intpr1` to print a scalar variable (`gfortran` 10 enforces the distinction between scalars and length-one arrays). Also `labelpr` to print just a label.
- `R_withCallingErrorHandler` is now available for establishing a calling handler in C code for conditions inheriting from class `error`.

Installation on a Unix-Alike:

- User-set '`DEFS`' (e.g., in '`config.site`') is now used for compiling packages (including base packages).
- There is a new variant option '`--enable-lto=check`' for checking consistency of BLAS/LAPACK/LINPACK calls — see 'Writing R Extensions'.
- A C++ compiler default is set only if the C++11 standard is supported: it no longer falls back to C++98.
- PCRE2 is used if available. To make use of PCRE1 if PCRE2 is unavailable, configure with option '`--with-pcre1`'.
- The minimum required version of `libcurl` is now 7.28.0 (Oct 2012).
- New make target `distcheck` checks

- R can be rebuilt from the tarball created by `make dist`,
- the build from the tarball passes `make check-all`,
- the build installs and uninstalls,
- the source files are properly cleaned by `make distclean`.

Utilities:

- `R --help` now mentions the option `--no-echo` (renamed from `--slave`) and its previously undocumented short form `-s`.
- `R CMD check` now optionally checks `configure` and `cleanup` scripts for non-Bourne-shell code ('bashisms').
- `R CMD check --as-cran` now runs `\donttest` examples (which are run by `example()`) instead of instructing the tester to do so. This can be temporarily circumvented during development by setting environment variable `_R_CHECK_DONTTEST_EXAMPLES_` to a false value.

Package Installation:

- There is the beginnings of support for the recently approved C++20 standard, specified analogously to C++14 and C++17. There is currently only limited support for this in compilers, with flags such as '`-std=c++20`' and '`-std=c++2a`'. For the time being the `configure` test is of accepting one of these flags and compiling C++17 code.

Bug Fixes:

- `formula(x)` with `length(x) > 1` character vectors, is deprecated now. Such use has been rare, and has 'worked' as expected in some cases only. In other cases, wrong `x` have silently been truncated, not detecting previous errors.
- Long-standing issue where the X11 device could lose events shortly after startup has been addressed (PR#16702).
- The `data.frame` method for `rbind()` no longer drops `<NA>` levels from factor columns by default (PR#17562).
- `available.packages()` and hence `install.packages()` now pass their `...` argument to `download.file()`, fulfilling the wish of PR#17532; subsequently, `available.packages()` gets new argument `quiet`, solving PR#17573.
- `stopifnot()` gets new argument `exprObject` to allow an R object of class `expression` (or other 'language') to work more consistently, thanks to suggestions by Suharto Anggono.
- `conformMethod()` now works correctly in cases containing a "`&&` logic" bug, reported by Henrik Bengtsson. It now creates methods with `"missing"` entries in the signature. Consequently, `rematchDefinition()` is amended to use appropriate `.local()` calls with named arguments where needed.
- `format.default(*, scientific = FALSE)` now corresponds to a practically most extreme `options(scipen = n)` setting rather than arbitrary `n = 100`.
- `format(as.symbol("foo"))` now works (returning `"foo"`).
- `postscript(.., title = *)` now signals an error when the title string contains a character which would produce corrupt PostScript, thanks to PR#17607 by Daisuko Ogawa.
- Certain `Ops` (notably comparison such as ==) now also work for 0-length data frames, after reports by Hilmar Berger.
- `methods(class = class(glm(..)))` now warns more usefully and only once.
- `write.dcf()` no longer mangles field names (PR#17589).

- Primitive replacement functions no longer mutate a referenced first argument when used outside of a complex assignment context.
- A better error message for `contour(*, levels = Inf)`.
- The return value of `contourLines()` is no longer `invisible()`.
- The Fortran code for calculating the `coefficients` component in `lm.influence()` was very inefficient. It has (for now) been replaced with much faster R code (PR#17624).
- `cm.colors(n)` *etc* no longer append the code for `alpha = 1`, `"FF"`, to all colors. Hence all eight `*.colors()` functions and `rainbow()` behave consistently and have the same non-explicit default (PR#17659).
- `dnorm` had a problematic corner case with `sd == -Inf` or negative `sd` which was not flagged as an error in all cases. Thanks to Stephen D. Weigand for reporting and Wang Jiefei for analyzing this; similar change has been made in `dlnorm()`.
- The optional `iter.smooth` argument of `plot.lm()`, (the `plot()` method for `lm` and `glm` fits) now defaults to `0` for all `glm` fits. Especially for binary observations with high or low fitted probabilities, this effectively deleted all observations of 1 or 0. Also, the type of residuals used in the `glm` case has been switched to `"pearson"` since deviance residuals do not in general have approximately zero mean.
- In `plot.lm`, Cook's distance was computed from unweighted residuals, leading to inconsistencies. Replaced with usual weighted version. (PR#16056)
- Time-series `ts(*, start, end, frequency)` with fractional `frequency` are supported more consistently; thanks to a report from Johann Kleinbub and analysis and patch by Duncan Murdoch in PR#17669.
- In case of errors `mcmapply()` now preserves attributes of returned `"try-error"` objects and avoids simplification, overriding `SIMPLIFY` to `FALSE`. (PR#17653)
- `as.difftime()` gets new optional `tz = "UTC"` argument which should fix behaviour during daylight-savings-changeover days, fixing PR#16764, thanks to proposals and analysis by Johannes Ranke and Kirill Müller.
- `round()` does a better job of rounding *"to nearest"* by *measuring* and *"to even"*; thanks to a careful algorithm originally prompted by the report from Adam Wheeler and then others, in PR#17668.
  `round(x, dig)` for *negative* digits is much more rational now, notably for large $|dig|$.
- Inheritance information on S4 classes is maintained more consistently, particularly in the case of class unions (in part due to PR#17596 and a report from Ezra Tucker).
- `is()` behaves more robustly when its argument `class2` is a `classRepresentation` object.
- The warning message when attempting to export an nonexistent class is now more readable; thanks to Thierry Onkelinx for recognizing the problem.
- `choose()` misbehaved in corner cases where it switched `n - k` for `k` and `n` was only *nearly* integer (report from Erik Scott Wright).
- `mle()` in the **stats4** package had problems combining use of box constraints and fixed starting values (in particular, confidence intervals were affected).
- Operator `?` now has lower precedence than `=` to work as documented, so `=` behaves like `<-` in help expressions (PR#16710).
- `smoothEnds(x)` now returns `integer` type in *both* cases when `x` is `integer`, thanks to a report and proposal by Bill Dunlap PR#17693.
- The **methods** package does a better job of tracking inheritance relationships across packages.
- `norm(diag(c(1, NA)), "2")` now works.

- `subset()` had problems with 0-col dataframes (reported by Bill Dunlap, PR#17721).
- Several cases of integer overflow detected by the 'undefined behaviour sanitizer' of `clang` 10 have been circumvented. One in `rhyper()` may change the generated value for large input values.
- `dotchart()` now places the y-axis label (`ylab`) much better, not overplotting labels, thanks to a report and suggestion by Alexey Shipunov.
- A rare C-level array overflow in `chull()` has been worked around.
- Some invalid specifications of the day-of-the-year (*via* `%j`, e.g. day 366 in 2017) or week plus day-of-the-week are now detected by `strptime()`. They now return `NA` but give a warning as they may have given random results or corrupted memory in earlier versions of R.
- `socketConnection(server = FALSE)` now respects the connection timeout also on Linux.
- `socketConnection(server = FALSE)` no longer leaks a connection that is available right away without waiting (e.g. on '`localhost`').
- Socket connections are now robust against spurious readability and spurious availability of an incoming connection.
- `blocking = FALSE` is now respected also on the server side of a socket connection, allowing non-blocking read operations.
- `anova.glm()` and `anova.glmlist()` computed incorrect score (Rao) tests in no-intercept cases. (André Gillibert, PR#17735)
- `summaryRprof()` now should work correctly for the `Rprof(*, memory.profiling=TRUE)` case with small chunk size (and `"tseries"` or similar) thanks to a patch proposal by Benjamin Tyner, in PR#15886.
- `xgettext()` ignores strings passed to `ngettext()`, since the latter is handled by `xngettext()`. Thanks to Daniele Medri for the report and all the recent work he has done on the Italian translations.
- `data(package = "P")` for `P` in **base** and **stats** no longer reports the data sets from package **datasets** (which it did for back compatibility for 16 years), fixing PR#17730.
- `x[[Inf]]` (returning `NULL`) no longer leads to undefined behavior, thanks to a report by Kirill Müller in PR#17756. Further, `x[[-Inf]]` and `x[[-n]]` now give more helpful error messages.
- `Gamma()` family sometimes had trouble storing link name PR#15891

Bug Fixes (Windows):

- `Sys.glob()` now supports all characters from the Unicode Basic Multilingual Plane, no longer corrupting some (less commonly used) characters (PR#17638).
- `Rterm` now correctly displays multi-byte-coded characters representable in the current native encoding (at least on Windows 10 they were sometimes omitted, PR#17632).
- `scan()` issues with UTF-8 data when running in a DBCS locale have been resolved (PR#16520, PR#16584).
- `Rterm` now accepts enhanced/arrow keys also with ConPTY.
- R can now be started *via* the launcher icon in a user documents directory whose path is not representable in the system encoding.
- `socketConnection(server = FALSE)` now returns instantly also on Windows when connection failure is signalled.
- Problems with UTF-16 surrogate pairs have been fixed in several functions, including `tolower()` and `toupper()` (PR#17645).

Changes in previous versions

- Older news can be found in text format in files 'NEWS.0', 'NEWS.1', 'NEWS.2' and 'NEWS.3' in the 'doc' directory. News in HTML format for R versions 3.x and from 2.10.0 to 2.15.3 are available at 'doc/html/NEWS.3.html' and 'doc/html/NEWS.2.html'.