:mod:`runpy` now uses :meth:`io.open_code` to open code files. Patch by Jason Killen.

Add additional audit events for the :mod:`ctypes` module.

Fixes audit event for :func:`os.system` to be named os.system.

Escape the server title of :class:`xmlrpc.server.DocXMLRPCServer` when rendering the document page as HTML. (Contributed by Dong-hee Na in :issue:`38243`.)

Update vendorized expat library version to 2.2.8, which resolves CVE-2019-15903.

Fixes email._header_value_parser.get_unstructured going into an infinite loop for a specific case in which the email header does not have trailing whitespace, and the case in which it contains an invalid encoded word. Patch by Ashwin Ramaswami.

Fix an infinite loop when parsing specially crafted email headers. Patch by Abhilash Raj.

Adds audit events for the range of supported run commands (see :ref:`using-on-general`).

ssl.match_hostname() no longer accepts IPv4 addresses with additional text after the address and only quad-dotted notation without trailing whitespaces. Some inet_aton() implementations ignore whitespace and all data after whitespace, e.g. '127.0.0.1 whatever'.

Adds audit events for :mod:`ensurepip`, :mod:`ftplib`, :mod:`glob`, :mod:`imaplib`, :mod:`nntplib`, :mod:`pdb`, :mod:`poplib`, :mod:`shutil`, :mod:`smtplib`, :mod:`sqlite3`, :mod:`subprocess`, :mod:`telnetlib`, :mod:`tempfile` and :mod:`webbrowser`, as well as :func:`os.listdir`, :func:`os.scandir` and :func:`breakpoint`.

Unknown interpreted text role "mod".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main)(Misc)(NEWS.d)3.9.0a1.rst`, **line 100**); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main)(Misc)(NEWS.d)3.9.0a1.rst`, **line 100**); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main)(Misc)(NEWS.d)3.9.0a1.rst`, **line 100**); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main)(Misc)(NEWS.d)3.9.0a1.rst`, **line 100**); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main)(Misc)(NEWS.d)3.9.0a1.rst`, **line 100**); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main)(Misc)(NEWS.d)3.9.0a1.rst`, **line 100**); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main)(Misc)(NEWS.d)3.9.0a1.rst`, **line 100**); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main)(Misc)(NEWS.d)3.9.0a1.rst`, **line 100**); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main)(Misc)(NEWS.d)3.9.0a1.rst`, **line 100**); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main)(Misc)(NEWS.d)3.9.0a1.rst`, **line 100**); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main)(Misc)(NEWS.d)3.9.0a1.rst`, **line 100**); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-`

:func:`io.open_code` is now used when reading :file:`.pth` files.

Updated OpenSSL to 1.1.1c in Windows installer

Fix parsing of invalid email addresses with more than one @ (e.g. a@b@c.com.) to not return the part before 2nd @ as valid email address. Patch by maxking & jpic.

Replace `Py_FatalError()` call with a regular :exc:`RuntimeError` exception in :meth:`float.__getformat__`.

Optimized :func:`math.floor()`, :func:`math.ceil()` and :func:`math.trunc()` for floats.

Fixed a bug in the compiler that was causing to raise in the presence of break statements and continue statements inside always false while loops. Patch by Pablo Galindo.

Optimized some set operations (e.g. |, ^, and -) of `dict_keys`. `d.keys() | other` was slower than `set(d) | other` but they are almost same performance for now.

`"".replace("", s, n)` now returns s instead of an empty string for all non-zero n. There are similar changes for :class:`bytes` and

:class:`bytearray` objects.

Fixed line numbers and column offsets for AST nodes for calls without arguments in decorators.

Fix a segmentation fault when using reverse iterators of empty `dict` objects. Patch by Dong-hee Na and Inada Naoki.

:class:`bytearray`, :class:`~array.array` and :class:`~mmap.mmap` objects allow now to export more than `2**31` buffers at a time.

Fixed a bug where the scope of named expressions was not being resolved correctly in the presence of the *global* keyword. Patch by Pablo Galindo.

Activate the `GC_DEBUG` macro for debug builds of the interpreter (when `Py_DEBUG` is set). Patch by Pablo Galindo.

When the garbage collector makes a collection in which some objects resurrect (they are reachable from outside the isolated cycles after the finalizers have been executed), do not block the collection of all objects that are still unreachable. Patch by Pablo Galindo and Tim Peters.

When cyclic garbage collection (gc) runs finalizers that resurrect unreachable objects, the current gc run ends, without collecting any cyclic trash. However, the statistics reported by `collect()` and `get_stats()` claimed that all cyclic trash found was collected, and that the resurrected objects were collected. Changed the stats to report that none were collected.

In debug mode, :c:func:`PyObject_GC_Track` now calls `tp_traverse()` of the object type to ensure that the object is valid: test that objects visited by `tp_traverse()` are valid.

Remove unnecessary intersection and update set operation in dictview with empty set. (Contributed by Dong-hee Na in :issue:`38210`.)

Check the error from the system's underlying `crypt` or `crypt_r`.

On FreeBSD, Python no longer calls `fedisableexcept()` at startup to control the floating point control mode. The call became useless since FreeBSD 6: it became the default mode.

Fix a bug due to the interaction of weakrefs and the cyclic garbage collector. We must clear any weakrefs in garbage in order to prevent their callbacks from executing and causing a crash.

Fix warnings options priority: `PyConfig.warnoptions` has the highest priority, as stated in the PEP 587.

Predict `BUILD_MAP_UNPACK_WITH_CALL` -> `CALL_FUNCTION_EX` opcode pairs in the main interpreter loop. Patch by Brandt Bucher.

Improve error handling for the assert_has_calls and assert_has_awaits methods of mocks. Fixed a bug where any errors encountered while binding the expected calls to the mock's spec were silently swallowed, leading to misleading error output.

Better control over symbol visibility is provided through use of the visibility attributes available in gcc >= 4.0, provided in a uniform way across POSIX and Windows. The POSIX build files have been updated to compile with -fvisibility=hidden, minimising exported symbols.

Optimized the :class:`dict` constructor and the :meth:`~dict.update` method for the case when the argument is a dict.

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d)3.9.0a1.rst`, line 367); *backlink***
>
> Unknown interpreted text role "class".

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d)3.9.0a1.rst`, line 367); *backlink***
>
> Unknown interpreted text role "meth".

Python now dumps path configuration if it fails to import the Python codecs of the filesystem and stdio encodings.

Allow to call `async_generator_athrow().throw(...)` even for non-started async generator helper. It fixes annoying warning at the end of :func:`asyncio.run` call.

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d)3.9.0a1.rst`, line 387); *backlink***
>
> Unknown interpreted text role "func".

Fix an off-by-one error in PyState_AddModule that could cause out-of-bounds memory access.

The select module is now PEP-384 compliant and no longer has static state

ast module updated to PEP-384 and all statics removed

The struct module is now PEP-384 compatible

The random module is now PEP-384 compatible

zlib module made PEP-384 compatible

Make pwd extension module PEP-384 compatible

grp module made PEP-384 compatible

Make _posixsubprocess PEP-384 compatible

Make termios extension module PEP-384 compatible

Fixed comparing and creating of InterpreterID and ChannelID.

Fix possible signed integer overflow when handling slices. Patch by hongweipeng.

Fixed silencing arbitrary errors if an attribute lookup fails in several sites. Only AttributeError should be silenced.

Optimize set difference_update for the case when the other set is much larger than the base set. (Suggested by Evgeny Kapun with code contributed by Michele OrrÃ¹).

The implementation of :func:`~unicodedata.is_normalized` has been greatly sped up on strings that aren't normalized, by implementing the full normalization-quick-check algorithm from the Unicode standard.

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d)3.9.0a1.rst`, line 529); *backlink***
>
> Unknown interpreted text role "func".

Adjust correctly the recursion level in the symtable generation for named expressions. Patch by Pablo Galindo.

The `CHECK_SMALL_INT` macro used inside :file:`Object/longobject.c` has been replaced with an explicit `return` at each call site.

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d)3.9.0a1.rst`, line 550); *backlink***

Unknown interpreted text role "file".

Fix :func:`codecs.lookup` to normalize the encoding name the same way than :func:`encodings.normalize_encoding`, except that :func:`codecs.lookup` also converts the name to lower case.

Fixed compilation of :keyword:`break` and :keyword:`continue` in the :keyword:`finally` block when the corresponding :keyword:`try` block contains :keyword:`return` with a non-constant value.

Improve import error message for partially initialized module on circular `from` imports - by Anthony Sottile.

Fix handling of negative indices in :c:member:`~PySequenceMethods.sq_item` of :class:`bytearray`. Patch by Sergey Fedoseev.

Slightly improve performance of :c:func:`PyLong_FromUnsignedLong`, :c:func:`PyLong_FromUnsignedLongLong` and :c:func:`PyLong_FromSize_t`. Patch by Sergey Fedoseev.

Ensure explicit relative imports from interactive sessions and scripts (having no parent package) always raise ImportError, rather than treating the current module as the package. Patch by Ben Lewis.

Reverted :issue:`32912`: emitting :exc:`SyntaxWarning` instead of :exc:`DeprecationWarning` for invalid escape sequences in string and bytes literals.

PEP 572: As described in the PEP, assignment expressions now raise :exc:`SyntaxError` when their interaction with comprehension scoping results in an ambiguous target scope.

The `TargetScopeError` subclass originally proposed by the PEP has been removed in favour of just raising regular syntax errors for the disallowed cases.

Fix potential use of uninitialized memory in :func:`os.wait3`.

Decoding bytes objects larger than 2GiB is faster and no longer fails when a multibyte characters spans a chunk boundary.

The :keyword:`assert` statement now works properly if the :exc:`AssertionError` exception is being shadowed. Patch by Zackery Spytz.

Unknown interpreted text role "exc".

Removed object cache (`free_list`) for bound method objects. Temporary bound method objects are less used than before thanks to the `LOAD_METHOD` opcode and the `_PyObject_VectorcallMethod` C API.

Fixed minor inconsistency in :meth:`list.__contains__`, :meth:`tuple.__contains__` and a few other places. The collection's item is now always at the left and the needle is on the right of ==.

Update differing exception between :meth:`builtins.__import__` and :meth:`importlib.__import__`.

When adding a wrapper descriptor from one class to a different class (for example, setting `__add__` = `str.__add__` on an `int` subclass), an exception is correctly raised when the operator is called.

Swap the positions of the *posonlyargs* and *args* parameters in the constructor of :class:`ast.parameters` nodes.

Optimized pymalloc for non PGO build.

Compute allocated pymalloc blocks inside _Py_GetAllocatedBlocks(). This slows down _Py_GetAllocatedBlocks() but gives a small speedup to _PyObject_Malloc() and _PyObject_Free().

Fix :func:`sys.excepthook` and :c:func:`PyErr_Display` if a filename is a bytes string. For example, for a SyntaxError exception where the filename attribute is a bytes string.

Fix `SyntaxError` indicator printing too many spaces for multi-line strings - by Anthony Sottile.

:meth:`bytearray.extend` now correctly handles errors that arise during iteration. Patch by Brandt Bucher.

The undocumented `sys.callstats()` function has been removed. Since Python 3.7, it was deprecated and always returned `None`.

It required a special build option `CALL_PROFILE` which was already removed in Python 3.7.

Remove `sys.getcheckinterval()` and `sys.setcheckinterval()` functions. They were deprecated since Python 3.2. Use :func:`sys.getswitchinterval` and :func:`sys.setswitchinterval` instead. Remove also `check_interval` field of the `PyInterpreterState` structure.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 794); *backlink*
>
> Unknown interpreted text role "func".

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 794); *backlink*
>
> Unknown interpreted text role "func".

In development mode and in debug build, *encoding* and *errors* arguments are now checked on string encoding and decoding operations. Examples: :func:`open`, :meth:`str.encode` and :meth:`bytes.decode`.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 806); *backlink*
>
> Unknown interpreted text role "func".

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 806); *backlink*
>
> Unknown interpreted text role "meth".

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 806); *backlink*
>
> Unknown interpreted text role "meth".

By default, for best performances, the *errors* argument is only checked at the first encoding/decoding error, and the *encoding* argument is sometimes ignored for empty strings.

Optimized decoding short ASCII string with UTF-8 and ascii codecs. `b"foo".decode()` is about 15% faster. Patch by Inada Naoki.

Improved support of the surrogatepass error handler in the UTF-8 and UTF-16 incremental decoders.

:func:`open`, :func:`io.open`, :func:`codecs.open` and :class:`fileinput.FileInput` no longer accept `'U'` ("universal newline") in the file mode. This flag was deprecated since Python 3.3.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 841); *backlink*
>
> Unknown interpreted text role "func".

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 841); *backlink*
>
> Unknown interpreted text role "func".

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 841); *backlink*
>
> Unknown interpreted text role "func".

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 841); *backlink*
>
> Unknown interpreted text role "class".

Reverse evaluation order of key: value in dict comprehensions as proposed in PEP 572. I.e. in `{k: v for ...}`, `k` will be evaluated before `v`.

Fix the :c:func:`PySys_Audit` call in :class:`mmap.mmap`.

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main)(Misc)(NEWS.d)3.9.0a1.rst`, line 862);** *backlink*
>
> Unknown interpreted text role "c:func".

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main)(Misc)(NEWS.d)3.9.0a1.rst`, line 862);** *backlink*
>
> Unknown interpreted text role "class".

Remove an unnecessary Py_XINCREF in classobject.c.

Fix a bug in the peephole optimizer that was not treating correctly constant conditions with binary operators. Patch by Pablo Galindo.

Python now gets the absolute path of the script filename specified on the command line (ex: "python3 script.py"): the __file__ attribute of the __main__ module and sys.path[0] become an absolute path, rather than a relative path.

Python's small object allocator (`obmalloc.c`) now allows (no more than) one empty arena to remain available for immediate reuse, without returning it to the OS. This prevents thrashing in simple loops where an arena could be created and destroyed anew on each iteration.

The dispatching of type slots to special methods (for example calling `__mul__` when doing `x * y`) has been made faster.

Implemented separate vectorcall functions for every calling convention of builtin functions and methods. This improves performance for calls.

Handle correctly negative line offsets in the peephole optimizer. Patch by Pablo Galindo.

Remove erroneous optimization for empty set differences.

Implement :c:func:`PyBuffer_SizeFromFormat()` function (previously documented but not implemented): call :func:`struct.calcsize`. Patch by Joannah Nanjekye.

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main)(Misc)(NEWS.d)3.9.0a1.rst`, line 953);** *backlink*
>
> Unknown interpreted text role "c:func".

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main)(Misc)(NEWS.d)3.9.0a1.rst`, line 953);** *backlink*
>
> Unknown interpreted text role "func".

Slot functions optimize any callable with `Py_TPFLAGS_METHOD_DESCRIPTOR` instead of only instances of `function`.

The slot `tp_vectorcall_offset` is inherited unconditionally to support `super().__call__()` when the base class uses vectorcall.

:func:`threading.get_native_id` now also supports NetBSD.

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main)(Misc)(NEWS.d)3.9.0a1.rst`, line 984);** *backlink*
>
> Unknown interpreted text role "func".

Add :func:`threading.get_native_id` support for AIX. Patch by M. Felt

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main)(Misc)(NEWS.d)3.9.0a1.rst`, line 993);** *backlink*
>
> Unknown interpreted text role "func".

:func:`sum` has been optimized for boolean values.

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main)(Misc)(NEWS.d)3.9.0a1.rst`, line 1002);** *backlink*
>
> Unknown interpreted text role "func".

Add `--upgrade-deps` to venv module. Patch by Cooper Ry Lees

`pdb.Pdb` supports ~/.pdbrc in Windows 7. Patch by Tim Hopper and Dan Lidral-Porter.

Updated encodings: - Removed the "tis260" encoding, which was an alias for the nonexistent "tactis" codec. - Added "mac_centeuro" as an alias for the mac_latin2 encoding.

The :class:`classmethod` decorator can now wrap other descriptors such as property objects. Adapted from a patch written by Graham Dumpleton.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, **line 1041**); *backlink*
>
> Unknown interpreted text role "class".

Improve speed of dictview intersection by directly using set intersection logic. Patch by David Su.

Prohibit parallel running of aclose() / asend() / athrow(). Fix ag_running to reflect the actual running status of the AG.

The :func:`curses.update_lines_cols` function now returns `None` instead of `1` on success.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, **line 1071**); *backlink*
>
> Unknown interpreted text role "func".

Update :exc:`TypeError` messages for :meth:`os.path.join` to include :class:`os.PathLike` objects as acceptable input types.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, **line 1081**); *backlink*
>
> Unknown interpreted text role "exc".

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, **line 1081**); *backlink*
>
> Unknown interpreted text role "meth".

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, **line 1081**); *backlink*
>
> Unknown interpreted text role "class".

Add a repr for `subprocess.Popen` objects. Patch by Andrey Doroschenko.

pydoc now recognizes and parses HTTPS URLs. Patch by python273.

Prevent asyncio from crashing if parent `__init__` is not called from a constructor of object derived from `asyncio.Future`.

:mod:`pdb` now uses :meth:`io.open_code` to trigger auditing events.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, **line 1119**); *backlink*
>
> Unknown interpreted text role "mod".

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, **line 1119**); *backlink*
>
> Unknown interpreted text role "meth".

Allow opening pipes and other non-seekable files in append mode with :func:`open`.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, **line 1128**); *backlink*
>
> Unknown interpreted text role "func".

Simplify the :mod:`argparse` usage message for `nargs="*"`.

WeakSet is now registered as a collections.abc.MutableSet.

logging: change RotatingHandler namer and rotator to class-level attributes. This stops __init__ from setting them to None in the case where a subclass defines them with eponymous methods.

Add :data:`os.P_PIDFD` constant, which may be passed to :func:`os.waitid` to wait on a Linux process file descriptor.

Add :class:`asyncio.PidfdChildWatcher`, a Linux-specific child watcher implementation that polls process file descriptors.

Expose the Linux `pidfd_open` syscall as :func:`os.pidfd_open`.

Added constants :data:`~fcntl.F_OFD_GETLK`, :data:`~fcntl.F_OFD_SETLK` and :data:`~fcntl.F_OFD_SETLKW` to the :mod:`fcntl` module. Patch by Dong-hee Na.

Fixed seeking backward on an encrypted :class:`zipfile.ZipExtFile`.

Add :func:`curses.get_escdelay`, :func:`curses.set_escdelay`, :func:`curses.get_tabsize`, and :func:`curses.set_tabsize` functions - by Anthony Sottile.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, **line 1216**); *backlink*
>
> Unknown interpreted text role "func".

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, **line 1216**); *backlink*
>
> Unknown interpreted text role "func".

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, **line 1216**); *backlink*
>
> Unknown interpreted text role "func".

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, **line 1216**); *backlink*
>
> Unknown interpreted text role "func".

Now :func:`~logging.config.fileConfig` correctly sets the .name of handlers loaded.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, **line 1227**); *backlink*
>
> Unknown interpreted text role "func".

Add new cache_parameters() method for functools.lru_cache() to better support pickling.

asynci.ProactorEventLoop.close() now only calls signal.set_wakeup_fd() in the main thread.

The case the result of :func:`pathlib.WindowsPath.glob` matches now the case of the pattern for literal parts.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, **line 1257**); *backlink*
>
> Unknown interpreted text role "func".

Remove misspelled attribute. The 3.8 changelog noted that this would be removed in 3.9.

Fixed erroneous equality comparison in statistics.NormalDist().

Added :data:`~os.CLD_KILLED` and :data:`~os.CLD_STOPPED` for :attr:`si_code`. Patch by Dong-hee Na.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, **line 1286**); *backlink*
>
> Unknown interpreted text role "data".

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, **line 1286**); *backlink*
>
> Unknown interpreted text role "data".

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, **line 1286**); *backlink*
>
> Unknown interpreted text role "attr".

Fixed a bug in :meth:`inspect.signature.bind` that was causing it to fail when handling a keyword argument with same name as positional-only parameter. Patch by Pablo Galindo.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, **line 1296**); *backlink*

Unknown interpreted text role "meth".

Fixed *hmac.new* and *hmac.HMAC* to raise TypeError instead of ValueError when the digestmod parameter, now required in 3.8, is omitted. Also clarified the hmac module documentation and docstrings.

Parameters *out* and *in* of :func:`os.sendfile` was renamed to *out_fd* and *in_fd*.

Added support for setting the umask in the child process to the subprocess module on POSIX systems.

Revert GH-15522, which introduces a regression in :meth:`mimetypes.guess_type` due to improper handling of filenames as urls.

Fix __repr__ method for :class:`dataclasses.InitVar` to support typing objects, patch by Samuel Colvin.

Add missing :data:`stat.S_IFDOOR`, :data:`stat.S_IFPORT`, :data:`stat.S_IFWHT`, :func:`stat.S_ISDOOR`, :func:`stat.S_ISPORT`, and :func:`stat.S_ISWHT` values to the Python implementation of :mod:`stat`.

Clarify docstrings of pathlib suffix(es)

Nested subclasses of :class:`typing.NamedTuple` are now pickleable.

Prevent :exc:`KeyError` thrown by :func:`_encoded_words.decode` when given an encoded-word with invalid content-type encoding from propagating all the way to :func:`email.message.get`.

Deprecated the `split()` method in :class:`_tkinter.TkappType` in favour of the `splitlist()` method which has more consistent and predicable behavior.

Add :exc:`smtplib.SMTPNotSupportedError` to the :mod:`smtplib` exported names.

sendfile() used in socket and shutil modules was raising OverflowError for files >= 2GiB on 32-bit architectures. (patch by Giampaolo Rodola)

Revert the new asyncio Streams API

OS native encoding is now used for converting between Python strings and Tcl objects. This allows to display, copy and paste to clipboard emoji and other non-BMP characters. Converting strings from Tcl to Python and back now never fails (except MemoryError).

Correctly handle pause/resume reading of closed asyncio unix pipe.

Child mocks will now detect their type as either synchronous or asynchronous, asynchronous child mocks will be AsyncMocks and synchronous child mocks will be either MagicMock or Mock (depending on their parent type).

Removes _AwaitEvent from AsyncMock.

Allow the rare code that wants to send invalid http requests from the *http.client* library a way to do so. The fixes for bpo-30458 led to breakage for some projects that were relying on this ability to test their own behavior in the face of bad requests.

Deprecate opening :class:`~gzip.GzipFile` for writing implicitly. Always specify the *mode* argument for writing.

Unknown interpreted text role "class".

Any synchronous magic methods on an AsyncMock now return a MagicMock. Any asynchronous magic methods on a MagicMock now return an AsyncMock.

Update the *length* parameter of :func:`os.pread` to accept :c:type:`Py_ssize_t` instead of :c:type:`int`.

:mod:`compileall` has a higher default recursion limit and new command-line arguments for path manipulation, symlinks handling, and multiple optimization levels.

asyncio: Fix inconsistent immediate Task cancellation

The arguments for the builtin pow function are more descriptive. They can now also be passed in as keywords.

Improve efficiency in parts of email package by changing while-pop to a for loop, using isdisjoint instead of set intersections.

Constructors of :class:`~typing.NamedTuple` and :class:`~typing.TypedDict` types now accept arbitrary keyword argument names, including "cls", "self", "typename", "_typename", "fields" and "_fields".

Add `__all__` to :mod:`datetime`. Patch by Tahia Khan.

Fixed case-insensitive string comparison in :class:`sqlite3.Row` indexing.

Changes AsyncMock call count and await count to be two different counters. Now await count only counts when a coroutine has been awaited, not when it has been called, and vice-versa. Update the documentation around this.

Fix default mock name in :meth:`unittest.mock.Mock.assert_called` exceptions. Patch by Abraham Toriz Cruz.

Fix a memory leak in comparison of :class:`sqlite3.Row` objects.

_hashlib no longer calls obsolete OpenSSL initialization function with OpenSSL 1.1.0+.

Preserve subclassing in inspect.Signature.from_callable.

Names of hashing algorithms from OpenSSL are now normalized to follow Python's naming conventions. For example OpenSSL uses sha3-512 instead of sha3_512 or blake2b512 instead of blake2b.

Fix a bug in dis.findlinestarts() where it would return invalid bytecode offsets. Document that a code object's co_lnotab can contain invalid bytecode offsets.

Add slots to :mod:`asyncio` transport classes, which can reduce memory usage.

The _hashlib OpenSSL wrapper extension module is now PEP-384 compliant.

hashlib constructors now support usedforsecurity flag to signal that a hashing algorithm is not used in a security context.

Fixes a potential incorrect AttributeError exception escaping ZipFile.extract() in some unsupported input error situations.

Remove obsolete copy of PBKDF2_HMAC_fast. All supported OpenSSL versions contain a fast implementation.

The OpenSSL hashlib wrapper uses a simpler implementation. Several Macros and pointless caches are gone. The hash name now comes from OpenSSL's EVP. The algorithm name stays the same, except it is now always lower case.

Fix parent class check in protocols to correctly identify the module that provides a builtin protocol, instead of assuming they all come from the :mod:`collections.abc` module

For :mod:`asyncio`, add a new coroutine :meth:`loop.shutdown_default_executor`. The new coroutine provides an API to schedule an executor shutdown that waits on the threadpool to finish closing. Also, :func:`asyncio.run` has been updated to utilize the new coroutine. Patch by Kyle Stanley.

Fixed regression bug for socket.getsockname() for non-CAN_ISOTP AF_CAN address family sockets by returning a 1-tuple instead of string.

Update parameter names on functions in importlib.metadata matching the changes in the 0.22 release of importlib_metadata.

The os.closewalk() implementation now uses the libc fdwalk() API on platforms where it is available.

Fixes AsyncMock so it doesn't crash when used with AsyncContextManagers or AsyncIterators.

Add warning to :meth:`datetime.utctimetuple`, :meth:`datetime.utcnow` and :meth:`datetime.utcfromtimestamp` .

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 1786); *backlink***
>
> Unknown interpreted text role "meth".

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 1786); *backlink***
>
> Unknown interpreted text role "meth".

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 1786); *backlink***
>
> Unknown interpreted text role "meth".

Allow passing a :term:`path-like object` as `directory` argument to the :class:`http.server.SimpleHTTPRequestHandler` class. Patch by GÃ©ry Ogam.

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 1796); *backlink***
>
> Unknown interpreted text role "term".

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 1796); *backlink***
>
> Unknown interpreted text role "class".

Update importlib.metadata with changes from importlib_metadata 0.21.

Remove __*code*__ check in AsyncMock that incorrectly evaluated function specs as async objects but failed to evaluate classes with __*await*__ but no __*code*__ attribute defined as async objects.

Fix reference counters in the :mod:`signal` module.

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 1827); *backlink***
>
> Unknown interpreted text role "mod".

Hide internal asyncio.Stream methods: feed_eof(), feed_data(), set_exception() and set_transport().

inspect.py now uses sys.exit() instead of exit()

Added command-line interface for the :mod:`ast` module.

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 1855); *backlink***
>
> Unknown interpreted text role "mod".

In :mod:`typing`, improved the __hash__ and __eq__ methods for :class:`ForwardReferences`.

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 1864); *backlink***
>
> Unknown interpreted text role "mod".

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 1864); *backlink***
>
> Unknown interpreted text role "class".

Fixed :func:`inspect.getattr_static` used `isinstance` while it should avoid dynamic lookup.

Update :class:`importlib.machinery.BuiltinImporter` to use `loader._ORIGIN` instead of a hardcoded value. Patch by Dong-hee Na.

In `importlib.metadata` sync with `importlib_metadata` 0.20, clarifying behavior of `files()` and fixing issue where only one requirement was returned for `requires()` on `dist-info` packages.

weakref.WeakValueDictionary defines a local remove() function used as callback for weak references. This function was created with a closure. Modify the implementation to avoid the closure.

Added the *indent* option to :func:`ast.dump` which allows it to produce a multiline indented output.

Fixed a crash in the :func:`tee` iterator when re-enter it. RuntimeError is now raised in this case.

Fix a ctypes regression of Python 3.8. When a ctypes.Structure is passed by copy to a function, ctypes internals created a temporary object which had the side effect of calling the structure finalizer (__del__) twice. The Python semantics requires a finalizer to be called exactly once. Fix ctypes internals to no longer call the finalizer twice.

`_json.scanstring` is now up to 3x faster when there are many backslash escaped characters in the JSON string.

Prevent shutil.rmtree exception when built on non-Windows system without fd system call support, like older versions of macOS.

Semaphores and BoundedSemaphores can now release more than one waiting thread at a time.

Subscripts to the *unittest.mock.call* objects now receive the same chaining mechanism as any other custom attributes, so that the following usage no longer raises a *TypeError*:

call().foo().__getitem__('bar')

Patch by blhsing

Fix C compiler warning caused by distutils.ccompiler.CCompiler.has_function.

Add `F_GETPATH` command to :mod:`fcntl`.

`repr()` of buffered and text streams now silences only expected exceptions when get the value of "name" and "mode" attributes.

Add a `total_nframe` field to the traces collected by the tracemalloc module. This field indicates the original number of frames before it was truncated.

Most features of the subprocess module now work again in subinterpreters. Only *preexec_fn* is restricted in subinterpreters.

Fix the rusage implementation of time.process_time() to correctly report the sum of the system and user CPU time.

Fix :func:`ast.dump` when call with incompletely initialized node.

Unknown interpreted text role "func".

Restores instantiation of Windows IOCP event loops from the non-main thread.

Add default implementation of the :meth:`ast.NodeVisitor.visit_Constant` method which emits a deprecation warning and calls corresponding methody `visit_Num()`, `visit_Str()`, etc.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 2071); *backlink*
>
> Unknown interpreted text role "meth".

Update test_statistics.py to verify that the statistics module works well for both C and Python implementations. Patch by Dong-hee Na

Added a new status code to the http module: 451 UNAVAILABLE_FOR_LEGAL_REASONS

Fix a segmentation fault that appeared when comparing instances of `datetime.timezone` and `datetime.tzinfo` objects. Patch by Pablo Galindo.

Deprecate having random.seed() call hash on arbitrary types.

Add optional keyword argument `exit_on_error` for :class:`ArgumentParser`.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 2122); *backlink*
>
> Unknown interpreted text role "class".

The :mod:`faulthandler` module no longer allocates its alternative stack at Python startup. Now the stack is only allocated at the first faulthandler usage.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 2131); *backlink*
>
> Unknown interpreted text role "mod".

Fix a duplicated debug message when :meth:`smtplib.SMTP.connect` is called.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 2142); *backlink*
>
> Unknown interpreted text role "meth".

venv: Don't generate unset variable warning on deactivate.

Fix dataclasses.is_dataclass when given an instance that never raises AttributeError in __getattr__. That is, an object that returns something for __dataclass_fields__ even if it's not a dataclass.

Fix `socket` module's `socket.connect(address)` function being unable to establish connection in case of interrupted system call. The problem was observed on all OSes which `poll(2)` system call can take only non-negative integers and -1 as a timeout value.

Optimizations for Fraction.__hash__ suggested by Tim Peters.

Fix `faulthandler.register(chain=True)` stack. faulthandler now allocates a dedicated stack of `SIGSTKSZ*2` bytes, instead of just `SIGSTKSZ` bytes. Calling the previous signal handler in faulthandler signal handler uses more than `SIGSTKSZ` bytes of stack memory on some platforms.

Add C fastpath for statistics.NormalDist.inv_cdf() Patch by Dong-hee Na

Remove the deprecated method *threading.Thread.isAlive()*. Patch by Dong-hee Na.

Add Fraction.as_integer_ratio() to match the corresponding methods in bool, int, float, and decimal.

Add an xml.etree.ElementTree.indent() function for pretty-printing XML trees. Contributed by Stefan Behnel.

Fix :mod:`difflib` ? hint in diff output when dealing with tabs. Patch by Anthony Sottile.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 2243); *backlink*
>
> Unknown interpreted text role "mod".

In `zipfile.Path`, when adding implicit dirs, ensure that ancestral directories are added and that duplicates are excluded.

Renamed and documented *test.bytecode_helper* as *test.support.bytecode_helper*. Patch by Joannah Nanjekye.

Fix xgettext warnings in :mod:`argparse`.

:meth:`writelines` method of :class:`io.BytesIO` is now slightly faster when many small lines are passed. Patch by Sergey Fedoseev.

ensurepip now uses *importlib.resources.read_binary()* to read data instead of *pkgutil.get_data()*. Patch by Joannah Nanjekye.

Mark calendar.py helper functions as being private. The follows PEP 8 guidance to maintain the style conventions in the module and it addresses a known case of user confusion.

Add definition of THREAD_STACK_SIZE for AIX in Python/thread_pthread.h The default thread stacksize caused crashes with the default recursion limit Patch by M Felt

The logging.getLogger() API now returns the root logger when passed the name 'root', whereas previously it returned a non-root logger named 'root'. This could affect cases where user code explicitly wants a non-root logger named 'root', or instantiates a logger using logging.getLogger(__name__) in some top-level module called 'root.py'.

Fix the implementation of curses `addch(str, color_pair)`: pass the color pair to `setcchar()`, instead of always passing 0 as the color pair.

Fix performance regression on regular expression parsing with huge character sets. Patch by Yann Vaginay.

The function :c:func:`PyImport_GetModule` now ensures any module it returns is fully initialized. Patch by Joannah Nanjekye.

Fix IndexError in :mod:`email` package when trying to parse invalid address fields starting with :.

The :mod:`parser` module is deprecated and will be removed in future versions of Python.

Completing WSA* error codes in :mod:`socket`.

Fixed comparisons of :class:`datetime.timedelta` and :class:`datetime.timezone`.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main)(Misc)(NEWS.d)3.9.0a1.rst`, **line 2396);** *backlink*

Unknown interpreted text role "class".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main)(Misc)(NEWS.d)3.9.0a1.rst`, **line 2396);** *backlink*

Unknown interpreted text role "class".

Synchronize `importlib.metadata` with importlib_metadata 0.19, improving handling of EGG-INFO files and fixing a crash when entry point names contained colons.

Correct :func:`curses.unget_wch` error message. Patch by Anthony Sottile.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main)(Misc)(NEWS.d)3.9.0a1.rst`, **line 2418);** *backlink*

Unknown interpreted text role "func".

Add :meth:`is_relative_to` in :class:`PurePath` to determine whether or not one path is relative to another.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main)(Misc)(NEWS.d)3.9.0a1.rst`, **line 2427);** *backlink*

Unknown interpreted text role "meth".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main)(Misc)(NEWS.d)3.9.0a1.rst`, **line 2427);** *backlink*

Unknown interpreted text role "class".

Fixed :meth:`argparse.ArgumentParser.format_usage` for mutually exclusive groups. Patch by Andrew Nester.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main)(Misc)(NEWS.d)3.9.0a1.rst`, **line 2437);** *backlink*

Unknown interpreted text role "meth".

Let math.dist() accept coordinates as sequences (or iterables) rather than just tuples.

Fixed `__eq__`, `__lt__` etc implementations in some classes. They now return :data:`NotImplemented` for unsupported type of the other operand. This allows the other operand to play role (for example the equality comparison with :data:`~unittest.mock.ANY` will return `True`).

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main)(Misc)(NEWS.d)3.9.0a1.rst`, **line 2457);** *backlink*

Unknown interpreted text role "data".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main)(Misc)(NEWS.d)3.9.0a1.rst`, **line 2457);** *backlink*

Unknown interpreted text role "data".

Make Activate.ps1 Powershell script static to allow for signing it.

Update wheels bundled with ensurepip (pip 19.2.3 and setuptools 41.2.0)

Bring consistency to venv shell activation scripts by always using __VENV_PROMPT__.

Allowed the pure Python implementation of :class:`datetime.timezone` to represent sub-minute offsets close to minimum and maximum boundaries, specifically in the ranges (23:59, 24:00) and (-23:59, 24:00). Patch by Ngalim Siregar

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main)(Misc)(NEWS.d)3.9.0a1.rst`, **line 2497);** *backlink*

Unknown interpreted text role "class".

In :mod:`posix`, use `ttyname_r` instead of `ttyname` for thread safety.

Make internal attributes for statistics.NormalDist() private.

Fix *NonCallableMock._call_matcher* returning tuple instead of *_Call* object when *self._spec_signature* exists. Patch by Elizabeth Uselton

Make *from tkinter import* * import only the expected objects.

Adding a value error when an invalid value in passed to nargs Patch by Robert Leenders

Exceptions from :mod:`enum` now use the __qualname of the enum class in the exception message instead of the __name__.

Fix `IndexError` when parsing email headers with unexpectedly ending bare-quoted string value. Patch by Abhilash Raj.

Make json.loads faster for long strings. (Patch by Marco Paolini)

Recognize "UTF-8" as a valid value for LC_CTYPE in locale._parse_localename.

Return :exc:`NotImplemented` in Python implementation of __eq__ for :class:`~datetime.timedelta` and :class:`~datetime.time` when the other object being compared is not of the same type to match C implementation. Patch by Karthikeyan Singaravelan.

Record calls to parent when autospecced object is attached to a mock using :func:`unittest.mock.attach_mock`. Patch by Karthikeyan Singaravelan.

"python3 -m test -jN --timeout=TIMEOUT" now kills a worker process if it runs longer than *TIMEOUT* seconds.

Fix serialization of display name in originator or destination address fields with both encoded words and special chars.

Improve error reporting for corrupt zip files with bad zip64 extra data. Patch by Daniel Hillier.

pickle.loads() no longer raises TypeError when the buffers argument is set to None

Correct behavior for zipfile.Path.parent when the path object identifies a subdirectory.

Fix the .col_offset attribute of nested :class:`ast.BinOp` instances which had a too large value in some situations.

Fixes a possible hang when using a timeout on *subprocess.run()* while capturing output. If the child process spawned its own children

or otherwise connected its stdout or stderr handles with another process, we could hang after the timeout was reached and our child was killed when attempting to read final output from the pipes.

Fix :func:`multiprocessing.util.get_temp_dir` finalizer: clear also the 'tempdir' configuration of the current process, so next call to `get_temp_dir()` will create a new temporary directory, rather than reusing the removed temporary directory.

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 2689);** *backlink*
>
> Unknown interpreted text role "func".

The distutils `bdist_wininst` command is deprecated in Python 3.8, use `bdist_wheel` (wheel packages) instead.

When *Enum.__str__* is overridden in a derived class, the override will be used by *Enum.__format__* regardless of whether mixin classes are present.

http.client now enables TLS 1.3 post-handshake authentication for default context or if a cert_file is passed to HTTPSConnection.

Update vendorized expat version to 2.2.7.

SSLContext.post_handshake_auth = True no longer sets SSL_VERIFY_POST_HANDSHAKE verify flag for client connections. Although the option is documented as ignored for clients, OpenSSL implicitly enables cert chain validation when the flag is set.

:func:`os.sched_setaffinity` now correctly handles errors that arise during iteration over its `mask` argument. Patch by Brandt Bucher.

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 2752);** *backlink*
>
> Unknown interpreted text role "func".

The :func:`os.getcwdb` function now uses the UTF-8 encoding on Windows, rather than the ANSI code page: see PEP 529 for the rationale. The function is no longer deprecated on Windows.

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 2762);** *backlink*
>
> Unknown interpreted text role "func".

The sqlite3 module now raises TypeError, rather than ValueError, if operation argument type is not str: execute(), executemany() and calling a connection.

Fix IndexError in parsing a header value ending unexpectedly. Patch by Abhilash Raj.

The *dist* argument for statistics.quantiles() is now positional only. The current name doesn't reflect that the argument can be either a dataset or a distribution. Marking the parameter as positional avoids confusion and makes it possible to change the name later.

Fix a bug that was causing the :mod:`queue` module to fail if the accelerator module was not available. Patch by Pablo Galindo.

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 2806);** *backlink*
>
> Unknown interpreted text role "mod".

:mod:`pprint` now has support for :class:`types.SimpleNamespace`. Patch by Carl Bordum Hansen.

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 2816);** *backlink*
>
> Unknown interpreted text role "mod".

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 2816);** *backlink*
>
> Unknown interpreted text role "class".

An :class:`~argparse.ArgumentParser` with `allow_abbrev=False` no longer disables grouping of short flags, such as `-vv`, but only disables abbreviation of long flags as documented. Patch by Zac Hatfield-Dodds.

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 2826);** *backlink*

Unknown interpreted text role "class".

:func:`unittest.mock.call` now preserves the order of keyword arguments in repr output. Patch by Karthikeyan Singaravelan.

Fix error unpickling datetime.time objects from Python 2 with seconds>=24. Patch by Justin Blanchard.

Add formal support for UDPLITE sockets. Support was present before, but it is now easier to detect support with `hasattr(socket, 'IPPROTO_UDPLITE')` and there are constants defined for each of the values needed: :py:obj:`socket.IPPROTO_UDPLITE`, :py:obj:`UDPLITE_SEND_CSCOV`, and :py:obj:`UDPLITE_RECV_CSCOV`. Patch by Gabe Appleton.

Optimized `functools.partial` by using vectorcall.

:meth:`sqlite3.Connection.create_aggregate`, :meth:`sqlite3.Connection.create_function`, :meth:`sqlite3.Connection.set_authorizer`, :meth:`sqlite3.Connection.set_progress_handler` :meth:`sqlite3.Connection.set_trace_callback` methods lead to segfaults if some of these methods are called twice with an equal object but not the same. Now callbacks are stored more carefully. Patch by Aleksandr Balezin.

The *obj* argument of :func:`dataclasses.replace` is positional-only now.

Add the optional Linux SocketCAN Broadcast Manager constants, used as flags to configure the BCM behaviour, in the socket module. Patch by Karl Ding.

`HTMLParser.unescape` is removed. It was undocumented and deprecated since Python 3.4.

Add .webmanifest -> application/manifest+json to list of recognized file types and content type headers

`aifc.openfp()` alias to `aifc.open()`, `sunau.openfp()` alias to `sunau.open()`, and `wave.openfp()` alias to `wave.open()` have been removed. They were deprecated since Python 3.7.

Deprecated accepting floats with integral value (like `5.0`) in :func:`math.factorial`.

`_dummy_thread` and `dummy_threading` modules have been removed. These modules were deprecated since Python 3.7 which requires threading support.

Email with single part but content-type set to `multipart/*` doesn't raise AttributeError anymore.

Use threadpool for reading from file for sendfile fallback mode.

Fix asyncio sendfile support when sendfile sends extra data in fallback mode.

:func:`ctypes.create_unicode_buffer()` now also supports non-BMP characters on platforms with 16-bit :c:type:`wchar_t` (for example, Windows and AIX).

In a subinterpreter, spawning a daemon thread now raises an exception. Daemon threads were never supported in subinterpreters. Previously, the subinterpreter finalization crashed with a Python fatal error if a daemon thread was still running.

Allow pure Python implementation of :mod:`pickle` to work even when the C :mod:`_pickle` module is unavailable.

Fix :mod:`lzma`: module decompresses data incompletely. When decompressing a FORMAT_ALONE format file, and it doesn't have the end marker, sometimes the last one to dozens bytes can't be output. Patch by Ma Lin.

Fix :meth:`RobotFileParser.crawl_delay` and :meth:`RobotFileParser.request_rate` to return `None` rather than raise :exc:`AttributeError` when no relevant rule is defined in the robots.txt file. Patch by Rémi Lapeyre.

Change the format of feature_version to be a (major, minor) tuple.

Eliminate :exc:`RuntimeError` raised by :func:`asyncio.all_tasks()` if internal tasks weak set is changed by another thread during iteration.

:class:`_pyio.IOBase` destructor now does nothing if getting the `closed` attribute fails to better mimic :class:`_io.IOBase` finalizer.

Fix a race condition at Python shutdown when waiting for threads. Wait until the Python thread state of all non-daemon threads get deleted (join all non-daemon threads), rather than just wait until non-daemon Python threads complete.

Default values which cannot be represented as Python objects no longer improperly represented as `None` in function signatures.

Added `encoding` and `errors` keyword parameters to `logging.basicConfig`.

Ensure cookies with `expires` attribute are handled in :meth:`CookieJar.make_cookies`.

Fix an unintended ValueError from :func:`subprocess.run` when checking for conflicting *input* and *stdin* or *capture_output* and *stdout* or *stderr* args when they were explicitly provided but with *None* values within a passed in **kwargs* dict rather than as passed directly by name. Patch contributed by Rémi Lapeyre.

The exception message for `inspect.getfile()` now correctly reports the passed class rather than the builtins module.

Give math.perm() a one argument form that means the same as math.factorial().

For math.perm(n, k), let k default to n, giving the same result as factorial.

Converted _collections._count_elements to use the Argument Clinic.

Do not always create a :class:`collections.deque` in :class:`asyncio.Lock`.

Speed-up statistics.fmean() by switching from a function to a generator.

Remove `Enum._convert` method, deprecated in 3.8.

argparse._ActionsContainer.add_argument now throws error, if someone accidentally pass FileType class object instead of instance of FileType as *type* argument

The socket module now has the :func:`socket.send_fds` and :func:`socket.recv.fds` methods. Contributed by Joannah Nanjekye, Shinya Okano and Victor Stinner.

Support running asyncio subprocesses when execution event loop in a thread on UNIX.

Lengthy email headers with UTF-8 characters are now properly encoded when they are folded. Patch by Jeffrey Kintscher.

Fixed a bug in email parsing where a message with invalid bytes in content-transfer-encoding of a multipart message can cause an AttributeError. Patch by Andrew Donnellan.

pathlib.Path instance's rename and replace methods now return the new Path instance.

:class:`urllib.request.ProxyHandler` now lowercases the keys of the passed dictionary.

Fix :func:`repr` on empty :class:`ZipInfo` object. Patch by MickaÃ«l Schoentgen.

Email headers containing RFC2047 encoded words are parsed despite the missing whitespace, and a defect registered. Also missing trailing whitespace after encoded words is now registered as a defect.

Port test_datetime to VxWorks: skip zoneinfo tests on VxWorks

Add parser for Message-ID header and add it to default HeaderRegistry. This should prevent folding of Message-ID using RFC 2048 encoded words.

Ensure method signature is used instead of constructor signature of a class while asserting mock object against method calls. Patch by Karthikeyan Singaravelan.

posix.getgrouplist() now works correctly when the user belongs to NGROUPS_MAX supplemental groups. Patch by Jeffrey Kintscher.

Fix race condition in ThreadPoolExecutor when worker threads are created during interpreter shutdown.

Fix `UserString.encode()` to correctly return `bytes` rather than a `UserString` instance.

Deprecate xml.etree.ElementTree.Element.copy() in favor of copy.copy().

Patch by Gordon P. Hemsley

Fix infinite loop in email header folding logic that would be triggered when an email policy's max_line_length is not long enough to include the required markup and any values in the message. Patch by Paul Ganssle

Removed methods Element.getchildren(), Element.getiterator() and ElementTree.getiterator() and the xml.etree.cElementTree module.

Remove the old plistlib API deprecated in Python 3.4

distutils sorts source file lists so that Extension .so files build more reproducibly by default

Ignore `ValueError` from `signal` with `interaction` in non-main thread.

Added `user`, `group` and `extra_groups` parameters to the subprocess.Popen constructor. Patch by Patrick McLean.

Fix compile error when `_uuid` headers conflicting included.

Deprecate `smtpd.MailmanProxy` ready for future removal.

:attr:`shlex.shlex.punctuation_chars` is now a read-only property.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 3441); *backlink*
>
> Unknown interpreted text role "attr".

Add support for boolean actions like `--foo` and `--no-foo` to argparse. Patch contributed by RÃ©mi Lapeyre.

Fixes a bug in :mod:`cgi` module when a multipart/form-data request has no *Content-Length* header.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 3460); *backlink*
>
> Unknown interpreted text role "mod".

The abstract base classes in :mod:`collections.abc` no longer are exposed in the regular :mod:`collections` module.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 3470); *backlink*
>
> Unknown interpreted text role "mod".

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 3470); *backlink*
>
> Unknown interpreted text role "mod".

Distutils won't check for rpmbuild in specified paths only.

Division handling of PurePath now returns NotImplemented instead of raising a TypeError when passed something other than an instance of str or PurePath. Patch by Roger Aiudi.

:func:`binascii.a2b_base64` is now up to 2 times faster. Patch by Sergey Fedoseev.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 3500); *backlink*
>
> Unknown interpreted text role "func".

Add additional aliases for HP Roman 8. Patch by Michael Osipov.

Fix uuid.getnode() on platforms with '.' as MAC Addr delimiter as well fix for MAC Addr format that omits a leading 0 in MAC Addr values. Currently, AIX is the only know platform with these settings. Patch by Michael Felt.

Add :meth:`~pathlib.Path.readlink`. Patch by Girts Folkmanis.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, **line 3530**); *backlink*
>
> Unknown interpreted text role "meth".

Made :func:`urllib.parse.unquote()` accept bytes in addition to strings. Patch by Stein Karlsen.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, **line 3539**); *backlink*
>
> Unknown interpreted text role "func".

lib2to3 now recognizes expressions after `*` and `**` like in `f(*[] or [])`.

Update :func:`shutil.move` function to allow for Path objects to be used as source argument. Patch by Emily Morehouse and Maxwell "5.13b" McKinnon.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, **line 3559**); *backlink*
>
> Unknown interpreted text role "func".

Added __format__ to IPv4 and IPv6 classes. Always outputs a fully zero- padded string. Supports b/x/n modifiers (bin/hex/native format). Native format for IPv4 is bin, native format for IPv6 is hex. Also supports '#' and '_' modifiers.

Fix urllib.parse.urlparse() with numeric paths. A string like "path:80" is no longer parsed as a path but as a scheme ("path") and a path ("80").

Fixed non-deterministic behavior related to mimetypes extension mapping and module reinitialization.

Explicitly mention abc support in functools.singledispatch

Provides more details about the interaction between :c:func:`fork` and CPython's runtime, focusing just on the C-API. This includes cautions about where :c:func:`fork` should and shouldn't be called.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, **line 3610**); *backlink*
>
> Unknown interpreted text role "c:func".

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, **line 3610**); *backlink*
>
> Unknown interpreted text role "c:func".

Modernize :mod:`email` examples from %-formatting to f-strings.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, **line 3621**); *backlink*
>
> Unknown interpreted text role "mod".

Document the fact that :exc:`RuntimeError` is raised if :meth:`os.fork` is called in a subinterpreter.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, **line 3630**); *backlink*
>
> Unknown interpreted text role "exc".

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, **line 3630**); *backlink*
>
> Unknown interpreted text role "meth".

Add Brazilian Portuguese to the language switcher at Python Documentation website.

Add list of no-longer-escaped chars to re.escape documentation

Modernized the plistlib documentation

Fix example usage of :c:func:`PyModule_AddObject` to properly handle errors.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main)(Misc)(NEWS.d)3.9.0a1.rst`, **line 3668**); *backlink*
>
> Unknown interpreted text role "c:func".

Fix a dead link in the distutils API Reference.

Warn more strongly and clearly about pickle insecurity

Added a link to dateutil.parser.isoparse in the datetime.fromisoformat documentation. Patch by Paul Ganssle

Deprecate info(), geturl(), getcode() methods in favor of the headers, url, and status properties, respectively, for HTTPResponse and addinfourl. Also deprecate the code attribute of addinfourl in favor of the status attribute. Patch by Ashwin Ramaswami

Mention `frame.f_trace` in :func:`sys.settrace` docs.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main)(Misc)(NEWS.d)3.9.0a1.rst`, **line 3717**); *backlink*
>
> Unknown interpreted text role "func".

Make :c:func:`PyThreadState_DeleteCurrent` Internal.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main)(Misc)(NEWS.d)3.9.0a1.rst`, **line 3726**); *backlink*
>
> Unknown interpreted text role "c:func".

Beginning edits to Whatsnew 3.8

Stop recommending getopt in the tutorial for command line argument parsing and promote argparse.

Remove implementation-specific behaviour of how venv's Deactivate works.

Fix wording of arguments for :class:`Request` in :mod:`urllib.request`

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main)(Misc)(NEWS.d)3.9.0a1.rst`, **line 3763**); *backlink*
>
> Unknown interpreted text role "class".

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main)(Misc)(NEWS.d)3.9.0a1.rst`, **line 3763**); *backlink*
>
> Unknown interpreted text role "mod".

Add a brief note to indicate that any new `sys.implementation` required attributes must go through the PEP process.

Documented that :class:`mailbox.Maildir` constructor doesn't attempt to verify the maildir folder layout correctness. Patch by Sviatoslav Sydorenko.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main)(Misc)(NEWS.d)3.9.0a1.rst`, **line 3782**); *backlink*
>
> Unknown interpreted text role "class".

Fix *importlib* examples to insert any newly created modules via importlib.util.module_from_spec() immediately into sys.modules instead of after calling loader.exec_module().

Thanks to Benjamin Mintz for finding the bug.

Slash ('/') is now part of syntax.

Fix PyList_GetItem index description to include 0.

Replace the dead link to the Tkinter 8.5 reference by John Shipman, New Mexico Tech, with a link to the archive.org copy.

Added possible exceptions to the description of os.chdir().

Documented that in :meth:`datetime.datetime.strptime()`, the leading zero in some two-digit formats is optional. Patch by Mike Gleen.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d)3.9.0a1.rst`, **line 3842**); *backlink*
>
> Unknown interpreted text role "meth".

Add decompression pitfalls to zipfile module documentation.

In the documentation for difflib, a note was added explicitly warning that the results of SequenceMatcher's ratio method may depend on the order of the input strings.

Restructured the :mod:`datetime` docs in the interest of making them more user-friendly and improving readability. Patch by Brad Solomon.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d)3.9.0a1.rst`, **line 3872**); *backlink*
>
> Unknown interpreted text role "mod".

Make C-API docs clear about what the "main" interpreter is.

The documentation for decimal string formatting using the :*g* specifier has been updated to reflect the correct exponential notation cutoff point. Original patch contributed by Tuomas Suutari.

Document and test that `tempfile` functions may accept a :term:`path-like object` for the `dir` argument. Patch by Anthony Sottile.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d)3.9.0a1.rst`, **line 3902**); *backlink*
>
> Unknown interpreted text role "term".

Added a note about the intended use of code in .pth files.

Fix the Doc/Makefile regarding PAPER environment variable and PDF builds

Add documentation for tkinter modules

Fix test_communicate() of test_asyncio.test_subprocess: use `support.LONG_TIMEOUT` (5 minutes), instead of just 1 minute.

Add timeout constants to :mod:`test.support` :data:`~test.support.LOOPBACK_TIMEOUT`, :data:`~test.support.INTERNET_TIMEOUT`, :data:`~test.support.SHORT_TIMEOUT` and :data:`~test.support.LONG_TIMEOUT`.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d)3.9.0a1.rst`, **line 3949**); *backlink*
>
> Unknown interpreted text role "mod".

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d)3.9.0a1.rst`, **line 3949**); *backlink*
>
> Unknown interpreted text role "data".

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d)3.9.0a1.rst`, **line 3949**); *backlink*
>
> Unknown interpreted text role "data".

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d)3.9.0a1.rst`, **line 3949**); *backlink*
>
> Unknown interpreted text role "data".

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d)3.9.0a1.rst`, **line 3949**); *backlink*
>
> Unknown interpreted text role "data".

test.regrtest now uses process groups in the multiprocessing mode (-jN command line option) if process groups are available: if :func:`os.setsid` and :func:`os.killpg` functions are available.

Fix a race condition in test_asyncio.test_start_tls_server_1(). Previously, there was a race condition between the test main() function which replaces the protocol and the test ServerProto protocol which sends ANSWER once it gets HELLO. Now, only the test main() function is responsible to send data, ServerProto no longer sends data.

Fix `test_compileall.test_compile_dir_maxlevels()` on Windows without long path support: only create 3 subdirectories instead of between 20 and 100 subdirectories.

On timeout, regrtest no longer attempts to call `popen.communicate()` again: it can hang until all child processes using stdout and stderr pipes completes. Kill the worker process and ignores its output. Change also the faulthandler timeout of the main process from 1 minute to 5 minutes, for Python slowest buildbots.

Fix test_gdb for Link Time Optimization (LTO) builds.

test_ssl now handles disabled TLS/SSL versions better. OpenSSL's crypto policy and run-time settings are recognized and tests for disabled versions are skipped. Tests also accept more TLS minimum_versions for platforms that override OpenSSL's default with strict settings.

The private keys for test_ssl were encrypted with 3DES in traditional PKCS#5 format. 3DES and the digest algorithm of PKCS#5 are blocked by some strict crypto policies. Use PKCS#8 format with AES256 encryption instead.

test.support now has a helper function to check for availability of a hash digest function. Several tests are refactored avoid MD5 and use SHA256 instead. Other tests are marked to use MD5 and skipped when MD5 is disabled.

Multiprocessing test test_mymanager() now also expects -SIGTERM, not only exitcode 0. BaseManager._finalize_manager() sends SIGTERM to the manager process if it takes longer than 1 second to stop, which happens on slow buildbots.

Multiprocessing tests: increase test_queue_feeder_donot_stop_onexc() timeout from 1 to 60 seconds.

Test with OpenSSL 1.1.1d

Increase code coverage for multiprocessing.shared_memory.

Add tests for json.dump(..., skipkeys=True). Patch by Dong-hee Na.

Enhance regrtest multiprocess timeout: write a message when killing a worker process, catch popen.kill() and popen.wait() exceptions, put a timeout on the second call to popen.communicate().

Add tests for ROT-13 codec.

Added tests for PyDateTime_xxx_GET_xxx() macros of the C API of the :mod:`datetime` module. Patch by Joannah Nanjekye.

Fix test_shared_memory_cleaned_after_process_termination name handling

Add :func:`test.support.catch_threading_exception`: context manager catching :class:`threading.Thread` exception using :func:`threading.excepthook`.

test_concurrent_futures now explicitly stops the ForkServer instance if it's running.

multiprocessing tests now stop the ForkServer instance if it's running: close the "alive" file descriptor to ask the server to stop and then remove its UNIX address.

test_distutils.test_build_ext() is now able to remove the temporary directory on Windows: don't import the newly built C extension ("xx") in the current process, but test it in a separated process.

test_concurrent_futures now cleans up multiprocessing to remove immediately temporary directories created by multiprocessing.util.get_temp_dir().

test_winconsoleio doesn't leak a temporary file anymore: use tempfile.TemporaryFile() to remove it when the test completes.

multiprocessing tests now explicitly call `_run_finalizers()` to immediately remove temporary directories created by tests.

urllib.request tests now call :func:`~urllib.request.urlcleanup` to remove temporary files created by `urlretrieve()` tests and to clear the `_opener` global variable set by `urlopen()` and functions calling indirectly `urlopen()`.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 4212); *backlink*
>
> Unknown interpreted text role "func".

Remove `Lib/test/outstanding_bugs.py`.

Fix test failures when IPv6 is unavailable or disabled.

Replace deprecated method "random.choose" with "random.choice" in "test_pkg_import.py".

Remove no longer necessary code from c locale coercion tests

Fix test_shutil to no longer leak temporary files.

Fix test_wsgiref.testEnviron() to no longer depend on the environment variables (don't fail if "X" variable is set).

Fix test_os.test_chown(): use os.getgroups() rather than grp.getgrall() to get groups. Rename also the test to test_chown_gid().

Add --cleanup option to python3 -m test to remove `test_python_*` directories of previous failed jobs. Add "make cleantest" to run `python3 -m test --cleanup`.

test_gdb no longer fails if it gets an "unexpected" message on stderr: it now ignores stderr. The purpose of test_gdb is to test that python-gdb.py commands work as expected, not to test gdb.

Avoid TimeoutError in test_asyncio: test_start_tls_server_1()

Fix test_asyncio ProactorLoopCtrlC: join the thread to prevent leaking a running thread and leaking a reference.

Fix :func:`test.support.catch_unraisable_exception`: its __exit__() method now ignores unraisable exception raised when clearing its `unraisable` attribute.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 4331); *backlink*
>
> Unknown interpreted text role "func".

regrtest now uses :func:`sys.unraisablehook` to mark a test as "environment altered" (ENV_CHANGED) if it emits an "unraisable exception". Moreover, regrtest logs a warning in this case.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 4342); *backlink*
>
> Unknown interpreted text role "func".

Use `python3 -m test --fail-env-changed` to catch unraisable exceptions in tests.

Fix assertions in `test_close` and `test_events_mask_overflow` devpoll tests.

Rewrite `_PyObject_IsFreed()` unit tests.

`test_venv.test_multiprocessing()` now explicitly calls `pool.terminate()` to wait until the pool completes.

Make test_ssl pass with LibreSSL. LibreSSL handles minimum and maximum TLS version differently than OpenSSL.

Make `test_source_encoding.test_issue2301` implementation independent. The test will work now for both CPython and IronPython.

Update `test.test_importlib.test_abc` to test `find_spec()`.

Modify the test_uuid logic to test when a program is available AND can be used to obtain a MACADDR as basis for an UUID. Patch by M. Felt

Fallback to a default reason when :func:`unittest.skip` is uncalled. Patch by Naitree Zhu.

On Windows, build scripts will now recognize and use python.exe from an active virtual env.

Fix _hashlib build when Blake2 is disabled, but OpenSSL supports it.

Misc/python-config.in now uses *getvar()* for all still existing *sysconfig.get_config_var()* calls. Patch by Joannah Nanjekye.

Fix stdatomic.h header check for ICC compiler: the ICC implementation lacks atomic_uintptr_t type which is needed by Python.

In Solaris family, we must be sure to use `-D_REENTRANT`. Patch by Jesús Cea Avión.

Locate `llvm-profdata` and `llvm-ar` binaries using `AC_PATH_TOOL` rather than `AC_PATH_TARGET_TOOL`.

The :file:`.gitignore` file systematically keeps "rooted", with a non-trailing slash, all the rules that are meant to apply to files in a specific place in the repo. Previously, when the intended file to ignore happened to be at the root of the repo, we'd most often accidentally also ignore files and directories with the same name anywhere in the tree.

The :file:`Tools/unicode/makeunicodedata.py` script, which is used for converting information from the Unicode Character Database into generated code and data used by the methods of :class:`str` and by the :mod:`unicodedata` module, now handles each character's data as a `dataclass` with named attributes, rather than a length-18 list of different fields.

The :file:`.gitignore` file no longer applies to any files that are in fact tracked in the Git repository. Patch by Greg Price.

Change "clean" makefile target to also clean the program guided optimization (PGO) data. Previously you would have to use "make clean" and "make profile-removal", or "make clobber".

Mark some individual tests to skip when --pgo is used. The tests marked increase the PGO task time significantly and likely don't help improve optimization of the final executable.

Reduce the number of unit tests run for the PGO generation task. This speeds up the task by a factor of about 15x. Running the full unit test suite is slow. This change may result in a slightly less optimized build since not as many code branches will be executed. If you are willing to wait for the much slower build, the old behavior can be restored using './configure [..] PROFILE_TASK="-m test --pgo-extended"'. We make no guarantees as to which PGO task set produces a faster build. Users who care should run their own relevant benchmarks as results can depend on the environment, workload, and compiler tool chain.

`make install` no longer installs `wininst-*.exe` files used by distutils bdist_wininst: bdist_wininst only works on Windows.

Many `PyRun_XXX()` functions like :c:func:`PyRun_String` were no longer exported in `libpython38.dll` by mistake. Export them

again to fix the ABI compatibility.

Enables use of SSE2 instructions in Windows 32-bit build.

Update optional extension module detection for AIX. ossaudiodev and spwd are not applicable for AIX, and are no longer reported as missing. 3rd-party packaging of ncurses (with ASIS support) conflicts with officially supported AIX curses library, so configure AIX to use libcurses.a. However, skip trying to build _curses_panel.

patch by M Felt

Fixes HTML Help shortcut when Windows is not installed to C drive

Ensure ntpath.realpath() correctly resolves relative paths.

Restores the internal C headers that were missing from the nuget.org and Microsoft Store packages.

Remove `pythonw.exe` dependency on the Microsoft C++ runtime.

Fix error message in activate.bat

Ensures `pyw.exe` launcher reads correct registry key.

Fixes `ntpath.realpath` failing on `sys.executable`.

Update bundled OpenSSL to 1.1.1d

Reduce overhead when using multiprocessing in a Windows virtual environment.

Allow py.exe launcher to locate installations from the Microsoft Store and improve display of active virtual environments.

The `pip.ini` is no longer included in the Nuget package.

Set Windows 8 as the minimum required version for API support

:func:`os.cpu_count` now returns active processors rather than maximum processors.

venv activate.bat now works when the existing variables contain double quote characters.

Prevent error calling :func:`os.path.realpath` on `'NUL'`.

Fix case sensitivity in test_pathlib and test_ntpath.

Fixes distutils not finding vcruntime140.dll with only the v142 toolset installed.

Ensure command-line and unattend.xml setting override previously detected states in Windows installer.

Fixes :func:`os.stat` failing for block devices on Windows

Fixes potential crash when calling :func:`os.readlink` (or indirectly through :func:`~os.path.realpath`) on a file that is not a supported link.

Improve the implementation of `winerror_to_errno()`.

:func:`os.dup` no longer fails for standard streams on Windows 7.

The `nul` file on Windows now returns True from :func:`~os.path.exists` and a valid result from :func:`os.stat` with `S_IFCHR` set.

Enable support for following symlinks in :func:`os.realpath`.

Treat all name surrogate reparse points on Windows in :func:`os.lstat` and other reparse points as regular files in :func:`os.stat`.

Add the module name in the formatted error message when DLL load fail happens during module import in `_PyImport_FindSharedFuncptrWindows()`. Patch by Srinivas Nyayapati.

Trying to import the :mod:`crypt` module on Windows will result in an :exc:`ImportError` with a message explaining that the module isn't supported on Windows. On other platforms, if the underlying `_crypt` module is not available, the ImportError will include a message explaining the problem.

Fixes the icons used for file associations to the Microsoft Store package.

Fix use of registry values to launch Python from Microsoft Store app.

Fix memory leak on Windows in creating an SSLContext object or running urllib.request.urlopen('https://...').

Switch Windows Store package's pip to use bundled :file:`pip.ini` instead of :envvar:`PIP_USER` variable.

Officially drop support for creating bdist_wininst installers on non-Windows systems.

Include the `FORMAT_MESSAGE_IGNORE_INSERTS` flag in `FormatMessageW()` calls.

Fixes path for :data:`sys.executable` when running from the Microsoft Store.

Don't collect unfinished processes with `subprocess._active` on Windows to cleanup later. Patch by Ruslan Kuprieiev.

Removes libpython38.a from standard Windows distribution.

Update Windows builds to use SQLite 3.28.0.

On Windows, :func:`os.dup` no longer creates an inheritable fd when handling a character file.

Ensure `time.tzname` is correct on Windows when the active code page is set to CP_UTF7 or CP_UTF8.

Make :data:`winreg.REG_MULTI_SZ` support zero-length strings.

Replace use of :c:func:`strcasecmp` for the system function :c:func:`_stricmp`. Patch by Minmin Gong.

Add native Bluetooth RFCOMM support to socket module.

Updated OpenSSL to 1.1.1d in macOS installer.

Move Azure Pipelines to latest VM versions and make macOS tests optional

Increase the default stack size of threads from 5MB to 16MB on macOS, to match the stack size of the main thread. This avoids crashes on deep recursion in threads.

Avoid test suite failures on macOS by no longer calling resource.setrlimit to increase the process stack size limit at runtime. The runtime change is no longer needed since the interpreter is being built with a larger default stack size.

Update macOS installer to use SQLite 3.28.0.

Updated OpenSSL to 1.1.1c in macOS installer.

Stop adding newline when saving an IDLE shell window.

Add an option to toggle IDLE's cursor blink for shell, editor, and output windows. See Settings, General, Window Preferences, Cursor Blink. Patch by Zackery Spytz.

Do not try to compile IDLE shell or output windows

IDLE no longer fails when write non-encodable characters to stderr. It now escapes them with a backslash, as the regular Python interpreter. Added the `errors` field to the standard streams.

When exiting IDLE, catch any AttributeError. One happens when EditorWindow.close is called twice. Printing a traceback, when IDLE is run from a terminal, is useless and annoying.

To avoid problems, test_idle ignores the user config directory. It no longer tries to create or access .idlerc or any files within. Users must run IDLE to discover problems with saving settings.

IDLE no longer adds 'argv' to the user namespace when initializing it. This bug only affected 3.7.4 and 3.8.0b2 to 3.8.0b4.

Shell restart lines now fill the window width, always start with '=', and avoid wrapping unnecessarily. The line will still wrap if the included file name is long relative to the width.

To avoid occasional spurious test_idle failures on slower machines, increase the `hover_delay` in test_tooltip.

Properly handle user input warnings in IDLE shell. Cease turning SyntaxWarnings into SyntaxErrors.

IDLE Settings dialog now closes properly when there is no shell window.

Add mousewheel scrolling for IDLE module, path, and stack browsers. Patch by George Zhang.

Fixed completions list appearing too high or low when shown above the current line.

Refactor IDLE autocomplete and improve testing.

Reorder the Run menu. Put the most common choice, Run Module, at the top.

Improve highlight config sample with example shell interaction and better labels for shell elements.

Settings dialog no longer expands with font size.

Initialize the Customize Run dialog with the command line arguments most recently entered before. The user can optionally edit before submitting them.

Fix code context not showing the correct context when first toggled on.

Optimize code context to reduce unneeded background activity. Font and highlight changes now occur along with text changes instead of after a random delay.

Cleanup `config.py` by inlining `RemoveFile` and simplifying the handling of `file` in `CreateConfigHandlers`.

Fix tab focus traversal order for help source and custom run dialogs.

Both subprocess connection error messages now refer to the 'Startup failure' section of the IDLE doc.

Add optional line numbers for IDLE editor windows. Windows open without line numbers unless set otherwise in the General tab of the configuration dialog.

To compensate for stack frames added by IDLE and avoid possible problems with low recursion limits, add 30 to limits in the user code execution process. Subtract 30 when reporting recursion limits to make this addition mostly transparent.

Properly 'attach' search dialogs to their main window so that they behave like other dialogs and do not get hidden behind their main window.

Adjust "Zoom Height" to individual screens by momentarily maximizing the window on first use with a particular screen. Changing screen settings may invalidate the saved height. While a window is maximized, "Zoom Height" has no effect.

Make calltip reminder about '/' meaning positional-only less obtrusive by only adding it when there is room on the first line.

Add 'Run... Customized' to the Run menu to run a module with customized settings. Any 'command line arguments' entered are added to sys.argv. One can suppress the normal Shell main module restart.

Gather Format menu functions into format.py. Combine paragraph.py, rstrip.py, and format methods from editor.py.

Update Valgrind suppression file to ignore a false alarm in :c:func:`PyUnicode_Decode` when using GCC builtin strcmp().

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d) 3.9.0a1.rst`, line 5378); *backlink***
>
> Unknown interpreted text role "c:func".

pathfix.py: Assume all files that end on '.py' are Python scripts when working recursively.

pdb's `--help` and `--version` long options now work.

Improve ArgumentClinic converter for floats.

Remove `Tools/scripts/h2py.py`: use cffi to access a C API in Python.

2to3 now works when run from a zipped standard library.

Argument Clinic now uses the argument name on errors with keyword-only argument instead of their position. Patch contributed by RÃ©mi Lapeyre.

Add option -k to pathscript.py script: preserve shebang flags. Add option -a to pathscript.py script: add flags.

Re-export some function compatibility wrappers for macros in `pythonrun.h`.

Provide :c:func:`Py_EnterRecursiveCall` and :c:func:`Py_LeaveRecursiveCall` as regular functions for the limited API. Previously, there were defined as macros, but these macros didn't work with the limited API which cannot access `PyThreadState.recursion_depth` field. Remove `_Py_CheckRecursionLimit` from the stable ABI.

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d)3.9.0a1.rst`, line 5463);** *backlink*
>
> Unknown interpreted text role "c:func".

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d)3.9.0a1.rst`, line 5463);** *backlink*
>
> Unknown interpreted text role "c:func".

The global variable :c:data:`PyStructSequence_UnnamedField` is now a constant and refers to a constant string.

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d)3.9.0a1.rst`, line 5476);** *backlink*
>
> Unknown interpreted text role "c:data".

Fixed possible leak in :c:func:`PyArg_Parse` and similar functions for format units `"es#"` and `"et#"` when the macro :c:macro:`PY_SSIZE_T_CLEAN` is not defined.

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d)3.9.0a1.rst`, line 5486);** *backlink*
>
> Unknown interpreted text role "c:func".

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d)3.9.0a1.rst`, line 5486);** *backlink*
>
> Unknown interpreted text role "c:macro".

Fix a crash in :class:`weakref.proxy` objects due to incorrect lifetime management when calling some associated methods that may delete the last reference to object being referenced by the proxy. Patch by Pablo Galindo.

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d)3.9.0a1.rst`, line 5497);** *backlink*
>
> Unknown interpreted text role "class".

The `_PyObject_CheckConsistency()` function is now also available in release mode. For example, it can be used to debug a crash in the `visit_decref()` function of the GC.

Revert the removal of PyThreadState_DeleteCurrent() with documentation.

Update audioop extension module to use the stable ABI (PEP-384). Patch by Tyler Kieft.

:c:func:`Py_SetPath` now sets :data:`sys.executable` to the program full path (:c:func:`Py_GetProgramFullPath`) rather than to the program name (:c:func:`Py_GetProgramName`).

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d)3.9.0a1.rst`, line 5538);** *backlink*
>
> Unknown interpreted text role "c:func".

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d)3.9.0a1.rst`, line 5538);** *backlink*

Unknown interpreted text role "data".

Python ignored arguments passed to :c:func:`Py_SetPath`, :c:func:`Py_SetPythonHome` and :c:func:`Py_SetProgramName`: fix Python initialization to use specified arguments.

The :c:func:`Py_UNREACHABLE` macro now calls :c:func:`Py_FatalError`.

Make dict and weakref offsets opaque for C heap types by passing the offsets through PyMemberDef

The C function `PyGen_NeedsFinalizing` has been removed. It was not documented, tested or used anywhere within CPython after the implementation of PEP 442. Patch by Joannah Nanjekye. (Patch by Joannah Nanjekye)

Options added by `PySys_AddXOption()` are now handled the same way than `PyConfig.xoptions` and command line `-X` options.

Fix a crash in `PySys_SetArgvEx(0, NULL, 0)`.

Fix subtype_dealloc to suppress the type decref when the base type is a C heap type

Add :c:func:`_PyObject_FunctionStr` to get a user-friendly string representation of a function-like object. Patch by Jeroen Demeyer.

The functions `PyEval_CallObject`, `PyEval_CallFunction`, `PyEval_CallMethod` and `PyEval_CallObjectWithKeywords` are deprecated. Use :c:func:`PyObject_Call` and its variants instead.

`PyCFunction_Call` is now a deprecated alias of :c:func:`PyObject_Call`.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d)3.9.0a1.rst`, line 5640); *backlink*
>
> Unknown interpreted text role "c:func".

The vectorcall protocol now requires that the caller passes only strings as keyword names.

The vectorcall protocol is now enabled for `type` objects: set `tp_vectorcall` to a vectorcall function to be used instead of `tp_new` and `tp_init` when calling the class itself.

Exclude Python-ast.h, ast.h and asdl.h from the limited API.

Add new function `_PyObject_CallOneArg` for calling an object with one positional argument.

Add :func:`PyConfig_SetWideStringList` function.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d)3.9.0a1.rst`, line 5689); *backlink*
>
> Unknown interpreted text role "func".

Add fast functions for calling methods: :c:func:`_PyObject_VectorcallMethod`, :c:func:`_PyObject_CallMethodNoArgs` and :c:func:`_PyObject_CallMethodOneArg`.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d)3.9.0a1.rst`, line 5698); *backlink*
>
> Unknown interpreted text role "c:func".

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d)3.9.0a1.rst`, line 5698); *backlink*
>
> Unknown interpreted text role "c:func".

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d)3.9.0a1.rst`, line 5698); *backlink*
>
> Unknown interpreted text role "c:func".

The :const:`METH_FASTCALL` calling convention has been documented.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d)3.9.0a1.rst`, line 5709); *backlink*
>
> Unknown interpreted text role "const".

The new function :c:func:`PyCode_NewWithPosOnlyArgs` allows to create code objects like :c:func:`PyCode_New`, but with an extra *posonlyargcount* parameter for indicating the number of positonal-only arguments.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d)3.9.0a1.rst`, line 5718); *backlink*
>
> Unknown interpreted text role "c:func".

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\(cpython-main) (Misc) (NEWS.d)3.9.0a1.rst`, line 5718); *backlink*
>
> Unknown interpreted text role "c:func".

Fix dtrace issue introduce by bpo-36842

Add a new public :c:func:`PyObject_CallNoArgs` function to the C API: call a callable Python object without any arguments. It is the most efficient way to call a callback without any argument. On x86-64, for example, `PyObject_CallFunctionObjArgs(func, NULL)` allocates 960 bytes on the stack per call, whereas `PyObject_CallNoArgs(func)` only allocates 624 bytes per call.

Fix the cast on error in :c:func:`PyLong_AsUnsignedLongLongMask()`.

Convert posixmodule.c statically allocated types `DirEntryType` and `ScandirIteratorType` to heap-allocated types.

Use singular/plural noun in error message when instantiating an abstract class with non-overriden abstract method(s).