Add `volatile` to the accumulator variable in `hmac.compare_digest`, making constant-time-defeating optimizations less likely.

`types.GenericAlias` will now raise a `TypeError` when attempting to initialize with a keyword argument. Previously, this would cause the interpreter to crash if the interpreter was compiled with debug symbols. This does not affect interpreters compiled for release. Patch by Ken Jin.

Several built-in and standard library types now ensure that their internal result tuples are always tracked by the :term:`garbage collector <garbage collection>`:

- :meth:`collections.OrderedDict.items() <collections.OrderedDict>`

- :meth:`dict.items`

- :func:`enumerate`

- :func:`functools.reduce`

- :func:`itertools.combinations`

- :func:`itertools.combinations_with_replacement`

- :func:`itertools.permutations`

Unknown interpreted text role "func".

- :func:`itertools.product`

  > **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\[cpython-main][Misc][NEWS.d]3.10.0a3.rst`, **line 47**); *backlink*
  >
  > Unknown interpreted text role "func".

- :func:`itertools.zip_longest`

  > **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\[cpython-main][Misc][NEWS.d]3.10.0a3.rst`, **line 49**); *backlink*
  >
  > Unknown interpreted text role "func".

- :func:`zip`

  > **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\[cpython-main][Misc][NEWS.d]3.10.0a3.rst`, **line 51**); *backlink*
  >
  > Unknown interpreted text role "func".

Previously, they could have become untracked by a prior garbage collection. Patch by Brandt Bucher.

Improve handling of exceptions near recursion limit. Converts a number of Fatal Errors in RecursionErrors.

PEP 626: After a return, the f_lineno attribute of a frame is always the last line executed.

Speed up comparison of bytes objects with non-bytes objects when option :option:`-b` is specified. Speed up comparison of bytarray objects with non-buffer object.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\[cpython-main][Misc][NEWS.d]3.10.0a3.rst`, **line 83**); *backlink*
>
> Unknown interpreted text role "option".

Port the `_warnings` extension module to the multi-phase initialization API ([PEP 489](#)). Patch by Victor Stinner.

On Windows, the `SIGINT` event, `_PyOS_SigintEvent()`, is now created even if Python is configured to not install signal handlers (if :c:member:`PyConfig.install_signal_handlers` equals to 0, or `Py_InitializeEx(0)`).

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\[cpython-main][Misc][NEWS.d]3.10.0a3.rst`, **line 104**); *backlink*
>
> Unknown interpreted text role "c:member".

Allow assignment expressions in set literals and set comprehensions as per PEP 572. Patch by Pablo Galindo.

Change function parameters annotations internal representation to tuple of strings. Patch provided by Yurii Karabas.

Fix a regression introduced by the new parser, where an unparenthesized walrus operator was not allowed within generator expressions.

Allow an unparenthesized walrus in subscript indexes.

Make sure that the compiler front-end produces a well-formed control flow graph. Be be more aggressive in the compiler back-end, as it is now safe to do so.

On Windows, fix a regression in signal handling which prevented to interrupt a program using CTRL+C. The signal handler can be run in a thread different than the Python thread, in which case the test deciding if the thread can handle signals is wrong.

:class:`types.GenericAlias` objects can now be the targets of weakrefs.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\[cpython-main][Misc][NEWS.d]3.10.0a3.rst`, **line 178**); *backlink*
>
> Unknown interpreted text role "class".

Optimise constant subexpressions that appear as part of named expressions (previously the AST optimiser did not descend into named expressions). Patch by Nick Coghlan.

Fixed a bug with the LOAD_ATTR opcode cache that was not respecting monkey-patching a class-level attribute to make it a descriptor. Patch by Pablo Galindo.

Convert :mod:`queue` to use heap types.

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\[cpython-main][Misc][NEWS.d]3.10.0a3.rst`, line 209);** *backlink*
>
> Unknown interpreted text role "mod".

Improved accuracy of line tracing events and f_lineno attribute of Frame objects. See PEP 626 for details.

Convert :mod:`mmap` to use heap types.

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\[cpython-main][Misc][NEWS.d]3.10.0a3.rst`, line 228);** *backlink*
>
> Unknown interpreted text role "mod".

Allow GenericAlias objects to use :ref:`union type expressions <types-union>`. This allows expressions like list[int] | dict[float, str] where previously a TypeError would have been thrown. This also fixes union type expressions not de-duplicating GenericAlias objects. (Contributed by Ken Jin in :issue:`42233`.)

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\[cpython-main][Misc][NEWS.d]3.10.0a3.rst`, line 237);** *backlink*
>
> Unknown interpreted text role "ref".

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\[cpython-main][Misc][NEWS.d]3.10.0a3.rst`, line 237);** *backlink*
>
> Unknown interpreted text role "issue".

The import system triggers a *ImportWarning* when it falls back to using *load_module()*.

CGIHTTPRequestHandler.run_cgi() HTTP_ACCEPT improperly parsed. Replace the special purpose getallmatchingheaders with generic get_all method and add relevant tests.

Original Patch by Martin Panter. Modified by Senthil Kumaran.

Fix issue when dis failed to parse function that has no line numbers. Patch provided by Yurii Karabas.

:func:`inspect.findsource` now raises :exc:`OSError` instead of :exc:`IndexError` when :attr:`co_lineno` of a code object is greater than the file length. This can happen, for example, when a file is edited after it was imported. PR by Irit Katriel.

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\[cpython-main][Misc][NEWS.d]3.10.0a3.rst`, line 283);** *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]3.10.0a3.rst`, line 283);** *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]3.10.0a3.rst`, line 283);** *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]3.10.0a3.rst`, line 283);** *backlink*
>
> Unknown interpreted text role "attr".

Fix handling of trailing comments by :func:`inspect.getsource`.

Remove unexpected call of `__bool__` when passing a `spec_arg` argument to a Mock.

Added itertools.pairwise()

Fix test_master_read() so that it succeeds on all platforms that either raise OSError or return b"" upon reading from master.

ChainMap.__iter__ no longer calls __getitem__ on underlying maps

:class:`~traceback.TracebackException` no longer holds a reference to the exception's traceback object. Consequently, instances of TracebackException for equivalent but non-equal exceptions now compare as equal.

Make test_openpty() avoid unexpected success due to number of rows and/or number of columns being == 0.

Remove loop parameter from `asyncio.subprocess` and `asyncio.tasks` functions. Patch provided by Yurii Karabas.

Remove loop parameter from `asyncio.open_connection` and `asyncio.start_server` functions. Patch provided by Yurii Karabas.

Add :func:`platform.freedesktop_os_release` function to parse freedesktop.org `os-release` files.

Removed the `formatter` module, which was deprecated in Python 3.4. It is somewhat obsolete, little used, and not tested. It was originally scheduled to be removed in Python 3.6, but such removals were delayed until after Python 2.7 EOL. Existing users should copy whatever classes they use into their code. Patch by Dong-hee Na and and Terry J. Reedy.

Deprecate zipimport.zipimporter.load_module() in favour of exec_module().

Updated tests for the pty library. test_basic() has been changed to test_openpty(); this additionally checks if slave termios and slave winsize are being set properly by pty.openpty(). In order to add support for FreeBSD, NetBSD, OpenBSD, and Darwin, this also adds test_master_read(), which demonstrates that pty.spawn() should not depend on an OSError to exit from its copy loop.

Remove loop parameter from `__init__` in all `asyncio.locks` and `asyncio.Queue` classes. Patch provided by Yurii Karabas.

Make :class:`filecmp.dircmp` respect subclassing. Now the :attr:`filecmp.dircmp.subdirs` behaves as expected when subclassing dircmp.

The exception :exc:`socket.timeout` is now an alias of :exc:`TimeoutError`.

Support signal module on VxWorks.

We fixed an issue in *pickle.whichmodule* in which importing *multiprocessing* could change the how pickle identifies which module an object belongs to, potentially breaking the unpickling of those objects.

Simplify the :mod:`importlib` external bootstrap code: `importlib._bootstrap_external` now uses regular imports to import builtin modules. When it is imported, the builtin :func:`__import__()` function is already fully working and so can be used to import builtin modules like :mod:`sys`. Patch by Victor Stinner.

<div style="border:1px solid black; padding:10px;">

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\[cpython-main][Misc][NEWS.d]3.10.0a3.rst`, **line 478**); *backlink*

Unknown interpreted text role "mod".

</div>

<div style="border:1px solid black; padding:10px;">

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\[cpython-main][Misc][NEWS.d]3.10.0a3.rst`, **line 478**); *backlink*

Unknown interpreted text role "func".

</div>

<div style="border:1px solid black; padding:10px;">

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\[cpython-main][Misc][NEWS.d]3.10.0a3.rst`, **line 478**); *backlink*

Unknown interpreted text role "mod".

</div>

Convert _sre module types to heap types (PEP 384). Patch by Erlend E. Aasland.

subprocess module update for DragonFlyBSD support.

Port the `_signal` extension module to the multi-phase initialization API ([PEP 489](#)). Patch by Victor Stinner and Mohamed Koubaa.

:func:`time.time()`, :func:`time.perf_counter()` and :func:`time.monotonic()` functions can no longer fail with a Python fatal error, instead raise a regular Python exception on failure.

<div style="border:1px solid black; padding:10px;">

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\[cpython-main][Misc][NEWS.d]3.10.0a3.rst`, **line 520**); *backlink*

Unknown interpreted text role "func".

</div>

<div style="border:1px solid black; padding:10px;">

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\[cpython-main][Misc][NEWS.d]3.10.0a3.rst`, **line 520**); *backlink*

Unknown interpreted text role "func".

</div>

<div style="border:1px solid black; padding:10px;">

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\[cpython-main][Misc][NEWS.d]3.10.0a3.rst`, **line 520**); *backlink*

Unknown interpreted text role "func".

</div>

Fixed :meth:`tkinter.ttk.Style.map`. The function accepts now the representation of the default state as empty sequence (as returned by `Style.map()`). The structure of the result is now the same on all platform and does not depend on the value of `wantobjects`.

<div style="border:1px solid black; padding:10px;">

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\[cpython-main][Misc][NEWS.d]3.10.0a3.rst`, **line 531**); *backlink*

Unknown interpreted text role "meth".

</div>

Fix various issues with `typing.Literal` parameter handling (flatten, deduplicate, use type to cache key). Patch provided by Yurii Karabas.

:func:`time.perf_counter()` on Windows and :func:`time.monotonic()` on macOS are now system-wide. Previously, they used an offset computed at startup to reduce the precision loss caused by the float type. Use :func:`time.perf_counter_ns()` and :func:`time.monotonic_ns()` added in Python 3.7 to avoid this precision loss.

<div style="border:1px solid black; padding:10px;">

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\[cpython-main][Misc][NEWS.d]3.10.0a3.rst`, **line 553**); *backlink*

Unknown interpreted text role "func".

</div>

Fixed support of non-BMP characters in :mod:`tkinter` on macOS.

Fix the :class:`threading.Thread` class at fork: do nothing if the thread is already stopped (ex: fork called at Python exit). Previously, an error was logged in the child process.

Port _ssl extension module to heap types.

The `onerror` callback from `shutil.rmtree` now receives correct function when `os.open` fails.

Fix *os.sendfile()* on illumos.

Add :data:`threading.__excepthook__` to allow retrieving the original value of :func:`threading.excepthook` in case it is set to a broken or a different value. Patch by Mario Corchero.

Implement PEP 451/spec methods on zipimport.zipimporter: find_spec(), create_module(), and exec_module().

This also allows for the documented deprecation of find_loader(), find_module(), and load_module().

Mock objects which are not unsafe will now raise an AttributeError if an attribute with the prefix asert, aseert, or assrt is accessed, in addition to this already happening for the prefixes assert or assret.

`sqlite3.OptimizedUnicode` has been undocumented and obsolete since Python 3.3, when it was made an alias to :class:`str`. It is now deprecated, scheduled for removal in Python 3.12.

Added :func:`threading.gettrace` and :func:`threading.getprofile` to retrieve the functions set by :func:`threading.settrace` and :func:`threading.setprofile` respectively. Patch by Mario Corchero.

Fixed writing binary Plist files larger than 4 GiB.

On Unix, the :func:`os.device_encoding` function now returns `'UTF-8'` rather than the device encoding if the :ref:`Python UTF-8 Mode <utf8-mode>` is enabled.

webbrowser: Ignore *NotADirectoryError* when calling `xdg-settings`.

Fix a stack overflow error for asyncio Task or Future repr().

The overflow occurs under some circumstances when a Task or Future recursively returns itself.

Improve asyncio.wait function to create the futures set just one time.

Update various modules in the stdlib to fall back on *__spec__.loader* when *__loader__* isn't defined on a module.

The *load_module()* methods found in importlib now trigger a DeprecationWarning.

Windows: Change `sysconfig.get_config_var('EXT_SUFFIX')` to the expected full `platform_tag.extension` format. Previously it was hard-coded to `.pyd`, now it is compatible with `distutils.sysconfig` and will result in something like `.cp38-win_amd64.pyd`. This brings windows into conformance with the other platforms.

The :func:`traceback.format_exception`, :func:`traceback.format_exception_only`, and :func:`traceback.print_exception` functions can now take an exception object as a positional-only argument.

Enum: fix regression involving inheriting a multiply-inherited enum

Convert :mod:`sqlite3` to use heap types (PEP 384). Patch by Erlend E. Aasland.

Added support for the XPath `!=` operator in xml.etree

Fix :meth:`pprint.PrettyPrinter.format` overrides being ignored for contents of small containers. The :func:`pprint._safe_repr` function was removed.

Expose the :c:func:`splice` as :func:`os.splice` in the :mod:`os` module. Patch by Pablo Galindo

Clarify the error message for :exc:`asyncio.IncompleteReadError` when `expected` is `None`.

Add async context manager support for contextlib.nullcontext.

:attr:`pathlib.PurePath.parents` now supports negative indexing. Patch contributed by Yaroslav Pankovych.

Added missing connect_accepted_socket() method to `asyncio.AbstractEventLoop`.

Extracting a symlink from a tarball should succeed and overwrite the symlink if it already exists. The fix is to remove the existing file or symlink before extraction. Based on patch by Chris AtLee, Jeffrey Kintscher, and Senthil Kumaran.

:mod:`urllib.request` and :mod:`http.client` now send `http/1.1` ALPN extension during TLS handshake when no custom context is supplied.

Add func:*os.eventfd* to provide a low level interface for Linux's event notification file descriptor.

Add AsyncContextDecorator to contextlib to support async context manager as a decorator.

Fix time-of-check/time-of-action issue in subprocess.Popen.send_signal.

Add an `is_async` identifier to :mod:`pyclbr`'s `Function` objects. Patch by Batuhan Taskaya

Add slice support to :attr:`pathlib.PurePath.parents`.

Tentative to deprecate `make suspicious` by first removing it from the CI and documentation builds, but keeping it around for manual uses.

Fix the URL for the IMAP protocol documents.

Language and version switchers, previously maintained in every cpython branches, are now handled by docsbuild-script.

Re-enable test_gdb on gdb 9.2 and newer: https://bugzilla.redhat.com/show_bug.cgi?id=1866884 bug is fixed in gdb 10.1.

Fix `test_asyncio.test_call_later()` race condition: don't measure asyncio performance in the `call_later()` unit test. The test failed randomly on the CI.

Fix test_netrc on VxWorks: create temporary directories using temp_cwd().

skip test_getaddrinfo_ipv6_scopeid_symbolic and test_getnameinfo_ipv6_scopeid_symbolic on VxWorks

skip test_test of test_mailcap on VxWorks

add shell requirement for test_pipes

skip some tests related to fifo on VxWorks

Fix test_doctest.py failures for VxWorks.

Include `_testinternalcapi` module in Windows installer for test suite

test_ssl: skip test_min_max_version_mismatch when TLS 1.0 is not available

Fix os module failures for VxWorks RTOS.

Fix fifo test cases for VxWorks RTOS.

remove libnet dependency from detect_socket() for VxWorks

Fix a race condition in "make regen-all" when make -jN option is used to run jobs in parallel. The clinic.py script now only use atomic write to write files. Moveover, generated files are now left unchanged if the content does not change, to not change the file modification time.

Fix building `pycore_bitutils.h` internal header on old clang version without `__builtin_bswap16()` (ex: Xcode 4.6.3 on Mac OS X 10.7). Patch by Joshua Root and Victor Stinner.

It is no longer possible to build the `_ctypes` extension module without :c:type:`wchar_t` type: remove `CTYPES_UNICODE` macro. Anyway, the :c:type:`wchar_t` type is required to build Python. Patch by Victor Stinner.

Support was removed for AIX 5.3 and below. See :issue:`40680`.

Addressed three compiler warnings found by undefined behavior sanitizer (ubsan).

Remove macro definition of `copysign` (to `_copysign`) in headers.

The Windows launcher now properly handles Python 3.10 when listing installed Python versions.

Fix build on macOS Big Sur when MACOSX_DEPLOYMENT_TARGET=11

Ensure distutils.unixxcompiler.find_library_file can find system provided libraries on macOS 11.

Add support for macOS 11 and Apple Silicon systems.

It is now possible to build "Universal 2" binaries using "--enable-universalsdk --with-universal-archs=universal2".

Binaries build on later macOS versions can be deployed back to older versions (tested up to macOS 10.9), when using the correct deployment target. This is tested using Xcode 11 and later.

Added Darwin specific madvise options to mmap module.

The `--enable-universalsdk` and `--with-universal-archs` options for the configure script now check that the specified architectures can be used.

Keep IDLE running on macOS. Remove obsolete workaround that prevented running files with shortcuts when using new universal2 installers built on macOS 11.

Fix reporting offset of the RE error in searchengine.

Get docstrings for IDLE calltips more often by using inspect.getdoc.

The smelly.py script now also checks the Python dynamic library and extension modules, not only the Python static library. Make also the script more verbose: explain what it does.

Allow :file:`Tools/i18n/pygettext.py` to detect calls to `gettext` in f-strings.

The :c:func:`PyType_FromSpecWithBases` and :c:func:`PyType_FromModuleAndSpec` functions now accept a single class as the *bases* argument.

Port :mod:`select` extension module to multiphase initialization (PEP 489).

Port _posixsubprocess extension module to multiphase initialization (PEP 489).

Port _posixshmem extension module to multiphase initialization (PEP 489)

Port _struct extension module to multiphase initialization (PEP 489)

Port :mod:`spwd` extension module to multiphase initialization (PEP 489)

main\Misc\NEWS.d\[cpython-main][Misc][NEWS.d]3.10.0a3.rst, line 1299); *backlink*

Unknown interpreted text role "mod".

Port :mod:`gc` extension module to multiphase initialization ([PEP 489](#))

Port _queue extension module to multiphase initialization ([PEP 489](#))

Convert :c:func:`Py_TYPE` and :c:func:`Py_SIZE` back to macros to allow using them as an l-value. Many third party C extension modules rely on the ability of using Py_TYPE() and Py_SIZE() to set an object type and size: `Py_TYPE(obj) = type;` and `Py_SIZE(obj) = size;`.

Port :mod:`symtable` extension module to multiphase initialization ([PEP 489](#))

Port :mod:`grp` and :mod:`pwd` extension modules to multiphase initialization ([PEP 489](#))

Port _random extension module to multiphase initialization ([PEP 489](#))

Port _hashlib extension module to multiphase initialization ([PEP 489](#))

Removed the undocumented `PyOS_InitInterrupts()` function. Initializing Python already implicitly installs signal handlers: see :c:member:`PyConfig.install_signal_handlers`. Patch by Victor Stinner.

The `Py_TRASHCAN_BEGIN` macro no longer accesses PyTypeObject attributes, but now can get the condition by calling the new private :c:func:`_PyTrash_cond()` function which hides implementation details.

:c:func:`Py_GetPath`, :c:func:`Py_GetPrefix`, :c:func:`Py_GetExecPrefix`, :c:func:`Py_GetProgramFullPath`, :c:func:`Py_GetPythonHome` and :c:func:`Py_GetProgramName` functions now return NULL if called before :c:func:`Py_Initialize`

(before Python is initialized). Use the new :ref:`Python Initialization Configuration API <init-config>` to get the :ref:`Python Path Configuration. <init-path-config>`. Patch by Victor Stinner.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\[cpython-main][Misc][NEWS.d]3.10.0a3.rst`, line 1398); *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]3.10.0a3.rst`, line 1398); *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]3.10.0a3.rst`, line 1398); *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]3.10.0a3.rst`, line 1398); *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]3.10.0a3.rst`, line 1398); *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]3.10.0a3.rst`, line 1398); *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]3.10.0a3.rst`, line 1398); *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]3.10.0a3.rst`, line 1398); *backlink*
>
> Unknown interpreted text role "ref".

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\[cpython-main][Misc][NEWS.d]3.10.0a3.rst`, line 1398); *backlink*
>
> Unknown interpreted text role "ref".

The :c:func:`PyConfig_Read` function now only parses :c:member:`PyConfig.argv` arguments once: :c:member:`PyConfig.parse_argv` is set to 2 after arguments are parsed. Since Python arguments are strippped from :c:member:`PyConfig.argv`, parsing arguments twice would parse the application options as Python options.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\[cpython-main][Misc][NEWS.d]3.10.0a3.rst`, line 1413); *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]3.10.0a3.rst`, line 1413); *backlink*
>
> Unknown interpreted text role "c:member".

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\[cpython-main][Misc][NEWS.d]3.10.0a3.rst`, line 1413); *backlink*

Unknown interpreted text role "c:member".

Added :c:func:`Py_NewRef` and :c:func:`Py_XNewRef` functions to increment the reference count of an object and return the object. Patch by Victor Stinner.

When :c:func:`Py_Initialize` is called twice, the second call now updates more :mod:`sys` attributes for the configuration, rather than only :data:`sys.argv`. Patch by Victor Stinner.

The :c:func:`PyType_FromModuleAndSpec` function now accepts NULL `tp_doc` slot.

Added :c:func:`PyModule_AddObjectRef` function: similar to :c:func:`PyModule_AddObject` but don't steal a reference to the value on success. Patch by Victor Stinner.

The :c:data:`METH_FASTCALL` calling convention is added to the limited API. The functions :c:func:`PyModule_AddType`, :c:func:`PyType_FromModuleAndSpec`, :c:func:`PyType_GetModule` and :c:func:`PyType_GetModuleState` are added to the limited API on Windows.

Unknown interpreted text role "c:data".

Add dedicated entry to PyAsyncMethods for sending values

:c:func:`PyType_GetSlot()` can now accept static types.

:c:func:`PyList_SET_ITEM`, :c:func:`PyTuple_SET_ITEM` and :c:func:`PyCell_SET` macros can no longer be used as l-value or r-value. For example, `x = PyList_SET_ITEM(a, b, c)` and `PyList_SET_ITEM(a, b, c) = x` now fail with a compiler error. It prevents bugs like `if (PyList_SET_ITEM (a, b, c) < 0)` ... test. Patch by Zackery Spytz and Victor Stinner.