

Show sys.version when -V option is supplied twice.

The with-statement now checks for `__enter__` before it checks for `__exit__`. This gives less confusing error messages when both methods are missing. Patch by Jonathan Ellington.

Fix the `set_inheritable()` file descriptor method on platforms that do not have the `ioctl` `FIOCLEX` and `FIONCLEX` commands.

Fix not getting the locale's charset upon initializing the interpreter, on platforms that do not have `langinfo`.

Fixed crash in `Py_DecodeLocale()` in debug build on Mac OS X when decode astral characters. Patch by Xiang Zhang.

Extra slash no longer added to `sys.path` components in case of empty compile-time `PYTHONPATH` components.

Improve speed of the `STORE_DEREF` opcode by 40%.

`PyDict_SetDefault` didn't combine split table when needed. Patch by Xiang Zhang.

Change `PendingDeprecationWarning` -> `DeprecationWarning`. As it was agreed in the issue, `__aiter__` returning an awaitable should result in `PendingDeprecationWarning` in 3.5 and in `DeprecationWarning` in 3.6.

Fix a reflink in code that raises `DeprecationWarning`.

Fix asynchronous generators `aclose()` and `athrow()` to handle `StopAsyncIteration` propagation properly.

Restored the `__reduce__()` methods of `datetime` objects.

Regular expression patterns, `_sre.SRE_Pattern` objects created by `re.compile()`, become comparable (only `x==y` and `x!=y` operators). This change should fix the issue #18383: don't duplicate warning filters when the `warnings` module is reloaded (thing usually only done in unit tests).

The `subprocess.Popen.wait` method's undocumented `endtime` parameter now raises a `DeprecationWarning`.

In ctypes, prevent a crash calling the `from_buffer()` and `from_buffer_copy()` methods on abstract classes like `Array`.

Makes `Path.resolve()` succeed on paths that do not exist. Patch by Vajrasky Kok

Fixed possible DoS and arbitrary code execution when handle plural form selections in the `gettext` module. The expression parser now supports exact syntax supported by GNU `gettext`.

Fixed possible crash in `_io.TextIOWrapper` deallocator when the garbage collector is invoked in other thread. Based on patch by Sebastian Cufre.

Optimize `loop.call_soon`.

Fix `get_event_loop()` return the current loop if called from coroutines/callbacks.

Fix `asyncio.isfuture()` to support `unittest.Mock`.

Fix reflink in `_asyncio.Future.__iter__().throw`.

Fix `inspect.isawaitable` to always return bool Patch by Justin Mayfield.

Make `loop` methods reject socket kinds they do not support.

Fix a reflink in `functools.lru_cache`.

Fix `asyncio.iscoroutinefunction` to handle `Mock` objects.

Fix `create_unix_server` to support `Path`-like objects (PEP 519).

Add `collections.abc.AsyncGenerator`.

Documented command-line interface of `zipfile`.

Now `test.support.rmtree` is able to remove unwritable or unreadable directories.

Various caches now are cleared before running every test file.

Fix out-of-tree building on AIX. Patch by Tristan Carel and Michael Haubenwallner.

Rename `--with-optimizations` to `--enable-optimizations`.

Prevent missing 'getentropy' declaration warning on macOS. Patch by Gareth Rees.