

A detailed painting of a scene featuring a window with green shutters. A ginger and white cat is sitting on the windowsill, looking out. The window is framed by a dense arrangement of yellow flowers and green leaves. A small wind chime hangs from the top of the window frame. The wall is white with some exposed brickwork. In the bottom right corner, there are more flowers, including pink and white ones. The overall style is soft and painterly.

# Python C API

Victor  
Stinner



# Victor Stinner

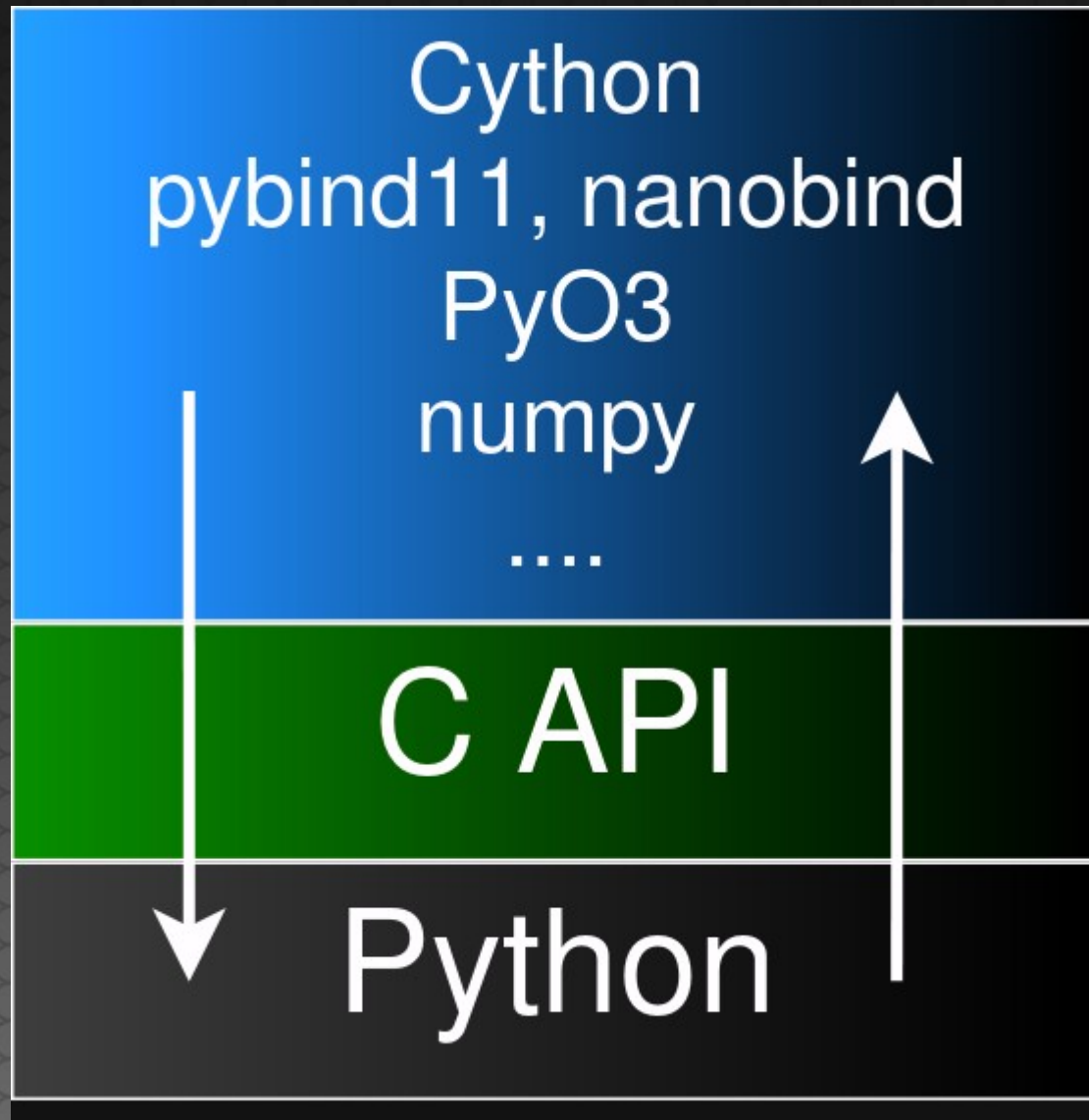
---



- CPython **core developer** for 14 years
- Maintain **Python CI** (GitHub Actions, buildbots), fix regressions, memory leaks, race conditions
- Work for **Red Hat**: backport fixes up to Python 3.6 and fix security vulnerabilities
- Happy **vim** and **Fedora** user!



# C API



Red Hat

# My C API long term goals



Limited C API by default!  
Stable ABI for all!

It's a long term goal,  
not the exact purpose of this talk



**Red Hat**



# Stable ABI

---



- **Build once**, work on all Python versions
- Less wheel packages to distribute (one per "platform")
- Platform: {OS, CPU, libc}
- Added to **Python 3.2** (2011) but is incomplete
- Used by **500+** C extensions on PyPI: PySide6, cryptography, ...

# MarkupSafe: 60 wheel pkgs



## Built Distributions

- [MarkupSafe-2.1.5-cp312-cp312-win\\_amd64.whl](#) (17.1 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.12 | Windows x86-64
- [MarkupSafe-2.1.5-cp312-cp312-win32.whl](#) (16.6 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.12 | Windows x86
- [MarkupSafe-2.1.5-cp312-cp312-musllinux\\_1\\_1\\_x86\\_64.whl](#) (33.0 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.12 | musllinux: musl 1.1+ x86-64
- [MarkupSafe-2.1.5-cp312-cp312-musllinux\\_1\\_1\\_i686.whl](#) (32.3 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.12 | musllinux: musl 1.1+ i686
- [MarkupSafe-2.1.5-cp312-cp312-musllinux\\_1\\_1\\_aarch64.whl](#) (33.9 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.12 | musllinux: musl 1.1+ ARM64
- [MarkupSafe-2.1.5-cp312-cp312-manylinux\\_2\\_17\\_x86\\_64.manylinux2014\\_x86\\_64.whl](#) (28.5 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.12 | manylinux: glibc 2.17+ x86-64
- [MarkupSafe-2.1.5-cp312-cp312-manylinux\\_2\\_17\\_aarch64.manylinux2014\\_aarch64.whl](#) (29.5 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.12 | manylinux: glibc 2.17+ ARM64
- [MarkupSafe-2.1.5-cp312-cp312-manylinux\\_2\\_5\\_i686.manylinux1\\_i686.manylinux\\_2\\_17\\_i686.manylinux2014\\_i686.whl](#) (27.9 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.12 | manylinux: glibc 2.17+ i686 | manylinux: glibc 2.5+ i686
- [MarkupSafe-2.1.5-cp312-cp312-macosx\\_10\\_9\\_x86\\_64.whl](#) (14.1 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.12 | macOS 10.9+ x86-64
- [MarkupSafe-2.1.5-cp312-cp312-macosx\\_10\\_9\\_universal2.whl](#) (18.2 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.12 | macOS 10.9+ universal2 (ARM64, x86-64)
- [MarkupSafe-2.1.5-cp311-cp311-win\\_amd64.whl](#) (17.2 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.11 | Windows x86-64
- [MarkupSafe-2.1.5-cp311-cp311-win32.whl](#) (16.7 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.11 | Windows x86
- [MarkupSafe-2.1.5-cp311-cp311-musllinux\\_1\\_1\\_x86\\_64.whl](#) (33.3 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.11 | musllinux: musl 1.1+ x86-64
- [MarkupSafe-2.1.5-cp311-cp311-musllinux\\_1\\_1\\_i686.whl](#) (32.7 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.11 | musllinux: musl 1.1+ i686
- [MarkupSafe-2.1.5-cp311-cp311-musllinux\\_1\\_1\\_aarch64.whl](#) (34.0 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.11 | musllinux: musl 1.1+ ARM64
- [MarkupSafe-2.1.5-cp311-cp311-manylinux\\_2\\_17\\_x86\\_64.manylinux2014\\_x86\\_64.whl](#) (28.2 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.11 | manylinux: glibc 2.17+ x86-64
- [MarkupSafe-2.1.5-cp311-cp311-manylinux\\_2\\_17\\_aarch64.manylinux2014\\_aarch64.whl](#) (29.0 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.11 | manylinux: glibc 2.17+ ARM64
- [MarkupSafe-2.1.5-cp311-cp311-manylinux\\_2\\_5\\_i686.manylinux1\\_i686.manylinux\\_2\\_17\\_i686.manylinux2014\\_i686.whl](#) (27.8 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.11 | manylinux: glibc 2.17+ i686 | manylinux: glibc 2.5+ i686
- [MarkupSafe-2.1.5-cp311-cp311-macosx\\_10\\_9\\_x86\\_64.whl](#) (14.1 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.11 | macOS 10.9+ x86-64
- [MarkupSafe-2.1.5-cp311-cp311-macosx\\_10\\_9\\_universal2.whl](#) (18.2 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.11 | macOS 10.9+ universal2 (ARM64, x86-64)
- [MarkupSafe-2.1.5-cp310-cp310-win\\_amd64.whl](#) (17.2 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.10 | Windows x86-64

- [MarkupSafe-2.1.5-cp39-cp39-win\\_amd64.whl](#) (17.2 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.9 | Windows x86-64
- [MarkupSafe-2.1.5-cp39-cp39-win32.whl](#) (16.7 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.9 | Windows x86
- [MarkupSafe-2.1.5-cp39-cp39-musllinux\\_1\\_1\\_x86\\_64.whl](#) (29.8 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.9 | musllinux: musl 1.1+ x86-64
- [MarkupSafe-2.1.5-cp39-cp39-musllinux\\_1\\_1\\_i686.whl](#) (29.5 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.9 | musllinux: musl 1.1+ i686
- [MarkupSafe-2.1.5-cp39-cp39-musllinux\\_1\\_1\\_aarch64.whl](#) (30.4 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.9 | musllinux: musl 1.1+ ARM64
- [MarkupSafe-2.1.5-cp39-cp39-manylinux\\_2\\_17\\_x86\\_64.manylinux2014\\_x86\\_64.whl](#) (25.7 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.9 | manylinux: glibc 2.17+ x86-64
- [MarkupSafe-2.1.5-cp39-cp39-manylinux\\_2\\_17\\_aarch64.manylinux2014\\_aarch64.whl](#) (26.5 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.9 | manylinux: glibc 2.17+ ARM64
- [MarkupSafe-2.1.5-cp39-cp39-manylinux\\_2\\_5\\_i686.manylinux1\\_i686.manylinux\\_2\\_17\\_i686.manylinux2014\\_i686.whl](#) (25.3 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.9 | manylinux: glibc 2.17+ i686 | manylinux: glibc 2.5+ i686
- [MarkupSafe-2.1.5-cp39-cp39-macosx\\_10\\_9\\_x86\\_64.whl](#) (14.1 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.9 | macOS 10.9+ x86-64
- [MarkupSafe-2.1.5-cp39-cp39-macosx\\_10\\_9\\_universal2.whl](#) (18.2 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.9 | macOS 10.9+ universal2 (ARM64, x86-64)
- [MarkupSafe-2.1.5-cp38-cp38-win\\_amd64.whl](#) (17.2 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.8 | Windows x86-64
- [MarkupSafe-2.1.5-cp38-cp38-win32.whl](#) (16.7 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.8 | Windows x86
- [MarkupSafe-2.1.5-cp38-cp38-musllinux\\_1\\_1\\_x86\\_64.whl](#) (29.9 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.8 | musllinux: musl 1.1+ x86-64
- [MarkupSafe-2.1.5-cp38-cp38-musllinux\\_1\\_1\\_i686.whl](#) (29.7 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.8 | musllinux: musl 1.1+ i686
- [MarkupSafe-2.1.5-cp38-cp38-musllinux\\_1\\_1\\_aarch64.whl](#) (30.5 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.8 | musllinux: musl 1.1+ ARM64
- [MarkupSafe-2.1.5-cp38-cp38-manylinux\\_2\\_17\\_x86\\_64.manylinux2014\\_x86\\_64.whl](#) (26.1 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.8 | manylinux: glibc 2.17+ x86-64
- [MarkupSafe-2.1.5-cp38-cp38-manylinux\\_2\\_17\\_aarch64.manylinux2014\\_aarch64.whl](#) (26.9 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.8 | manylinux: glibc 2.17+ ARM64
- [MarkupSafe-2.1.5-cp38-cp38-manylinux\\_2\\_5\\_i686.manylinux1\\_i686.manylinux\\_2\\_17\\_i686.manylinux2014\\_i686.whl](#) (25.8 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.8 | manylinux: glibc 2.17+ i686 | manylinux: glibc 2.5+ i686
- [MarkupSafe-2.1.5-cp38-cp38-macosx\\_10\\_9\\_x86\\_64.whl](#) (14.1 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.8 | macOS 10.9+ x86-64
- [MarkupSafe-2.1.5-cp38-cp38-macosx\\_10\\_9\\_universal2.whl](#) (18.2 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.8 | macOS 10.9+ universal2 (ARM64, x86-64)
- [MarkupSafe-2.1.5-cp37-cp37m-win\\_amd64.whl](#) (17.2 kB [view hashes](#))  
Uploaded Feb 2, 2024 | CPython 3.7m | Windows x86-64
- [MarkupSafe-2.1.5-cp37-cp37m-win32.whl](#) (16.6 kB [view hashes](#))



Red Hat

# PySide6: 4 wheel packages



## Source Distributions

! No source distribution files available for this release. See tutorial on [generating distribution archives](#).

## Built Distributions

-  [PySide6-6.7.2-cp39-abi3-win\\_amd64.whl](#) (538.0 kB [view hashes](#))  
Uploaded Jun 18, 2024 CPython 3.9+ Windows x86-64
-  [PySide6-6.7.2-cp39-abi3-manylinux\\_2\\_31\\_aarch64.whl](#) (530.7 kB [view hashes](#))  
Uploaded Jun 18, 2024 CPython 3.9+ manylinux: glibc 2.31+ ARM64
-  [PySide6-6.7.2-cp39-abi3-manylinux\\_2\\_28\\_x86\\_64.whl](#) (530.8 kB [view hashes](#))  
Uploaded Jun 18, 2024 CPython 3.9+ manylinux: glibc 2.28+ x86-64
-  [PySide6-6.7.2-cp39-abi3-macosx\\_11\\_0\\_universal2.whl](#) (530.2 kB [view hashes](#))  
Uploaded Jun 18, 2024 CPython 3.9+ macOS 11.0+ universal2 (ARM64, x86-64)



Red Hat



# My C API long term goals

---



- More **freedom** to change Python **internals**
- Reduce 3<sup>rd</sup> party **maintenance burden** when updating Python (goal: no change)
- Reduce **friction/stress** related to the C API, on the Python side, and on the 3<sup>rd</sup> party side
- Get C extensions ready since Python **alpha1** release: no need to request release



Red Hat



# Challenges

---



- **Need to change** some 3<sup>rd</sup> party code to reach this long term goal
- Usually **1-10 lines** per Python release in the few **impacted projects**
- **Unknown number** of impacted projects per change



**Red Hat**





Python 3.13



# C API Working Group

---



- **PEP 731** "C API Working Group Charter"
- Created in November 2023
- New C API must be approved by the WG
- 6 members: **Erlend** Aasland, **Michael** Droettboom, **Petr** Viktorin, **Serhiy** Storchaka, **Steve** Dower, **Victor** Stinner
- **PEP 733** "An Evaluation of Python's Public C API"



**Red Hat**

# C API WG projects

---



- **decisions**: decide on an API
- **api-evolution**: list small API changes
- **api-revolution**: list disruptive API changes

<https://github.com/capi-workgroup/>



**Red Hat**



# Remove private funcs

---



- No **doc**, no **test**
- No **backward compatibility** support
- Private functions are not part of the **limited C API**
- Replace them with **public functions**



Red Hat

# Remove private funcs

---



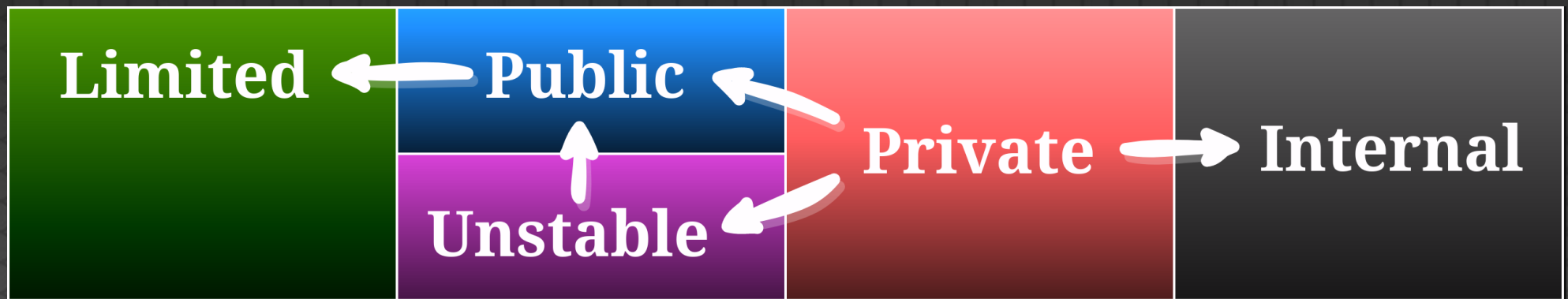
- Removed 300+ **private** functions in Python 3.13 alpha1
- Some removals impacted **5-30 projects**
- As planned, removals causing most trouble have been **reverted** (mostly in alpha2)
- 3.13 beta1: **264** functions removed



Red Hat



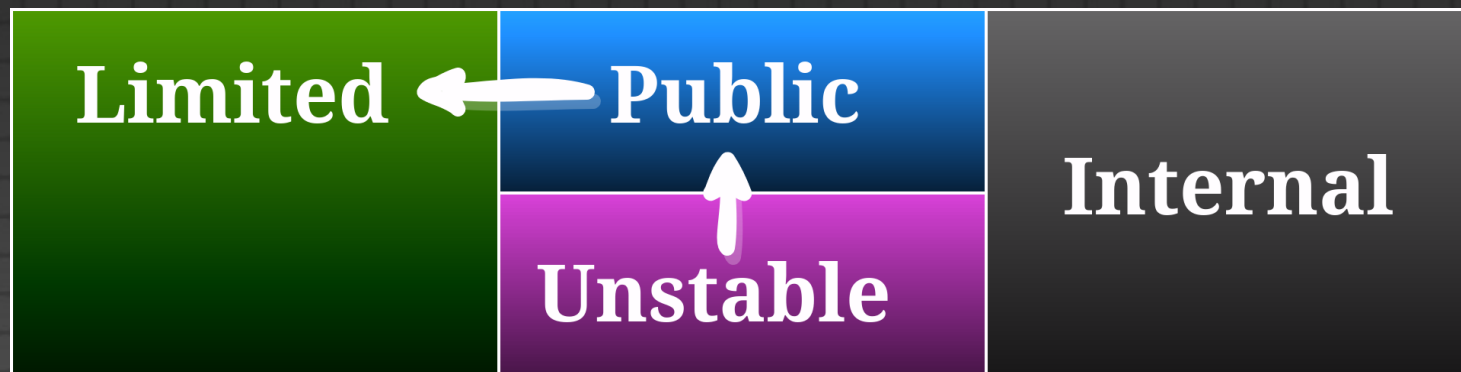
# Remove private funcs



Red Hat

# Remove private funcs

---



Red Hat



# Better public functions

---



- Promote private to **public** functions
- Add **backward compatibility warranty**
- Add **documentation** and **tests**
- C API Working Group designs a **better** API
- Enforce error checking
- Avoid inefficient `PyErr_Occurred()` check



Red Hat

# PyDict\_Pop() example

---



- 3.12: **PyObject\*** \_PyDict\_Pop(PyObject \*dict, PyObject \*key, PyObject \*default)  
→ NULL means "not found" **or** error
- 3.13: **int** PyDict\_Pop(PyObject \*op, PyObject \*key, **PyObject \*\*result**)
  - Return **-1** on **error** and set an exception
  - Return **0** if **not found** (NULL result)
  - Return **1** if **found** (non-NULL result)



Red Hat



# And Python <= 3.12 ?

---



**pythoncapi-compat** provides new Python 3.13 functions on Python 3.12 and older

- `PyDict_GetItemRef()`
- `PyLong_AsInt()`
- `PyUnicode_EqualToUTF8()`
- ...



**Red Hat**



# Python 3.13 Limited C API





# Argument Clinic Limited API

---



- Use limited C API if code uses **#define Py\_LIMITED\_API** ...
- Only use **public** functions of the limited C API
- **Inline code** to avoid calling private/internal functions



Red Hat

# AC: fcntl.fcntl() clinic input

---



```
fd: files  
cmd as code: int  
arg: object(c_default='NULL') = 0  
/
```



Red Hat



# AC:fcntl.fcntl() code in 3.12

---



```
if (!_PyArg_CheckPositional(  
    "fcntl", nargs, 2, 3)) ...  
if (!_PyLong_FileDescriptor_Converter(  
    args[0], &fd)) ...  
code = _PyLong_AsInt(args[1]);
```



Red Hat

# AC:fcntl.fcntl() code in 3.13

---



```
if (nargs < 2) { PyErr_Format(...)... }
if (nargs > 3) { PyErr_Format(...)... }
fd = PyObject_AsFileDescriptor(args[0]);
if (fd < 0) ...
code = PyLong_AsInt(args[1]);
```



Red Hat



# Build stdlib ext with limited

---



- Limited C API was not really used/tested by Python itself
- **Test** the limited C API
- Make sure that it **works** with non trivial code
- "Eating your own dog food"



Red Hat

# Build stdlib ext with limited

---



- **Build** 16 stdlib extensions with the **limited C API**
- errno, md5, resource, \_uuid, ...
- PEP 737 "Unify type formatting" adds **%T** and **%N** formats for error messages



Red Hat





Future  
(3.14)

# HPy

---



- New C API which looks like the Python C API
- **More efficient** on PyPy
- **"Universal mode"** which is a stable ABI working any CPython version and any PyPy version
- **numpy** is being porting to HPy



Red Hat



# Limited C API & stakeholders



- Cython: **opt-in** build mode
- PyO3 (Rust): **opt-in** build mode; consider making it the default!
- pybind11 (C++11): **not** supported
- nanobind (C++11): **opt-in** build mode, 100% feature complete
- cffi: **not** supported



**Red Hat**

# Use the Limited C API

---



- **CPython**: define `Py_LIMITED_API` macro
- **nanobind**: define `Py_LIMITED_API` macro
- **Cython**: define `CYTHON_LIMITED_API` macro

The macro must be set to a Python "hex" version.

Example: `0x03080000` for Python **3.8**.



**Red Hat**



# abi3audit

---



- Test if a binary uses the stable ABI or not
- <https://github.com/trailofbits/abi3audit>



Red Hat



Q & A

