



Python Native Interface Project: a Python C API revolution

Modern C API and ABI adapted for fast Python Implementations



Python JITs

To stay relevant for the next years, Python has to finally become much faster

- Zero-cost abstraction is possible (as with PyPy)
- Fast Python implementations exist (PyPy, GraalPy)
- CPython's JIT...



The problem: the current C CPython API

- Usage of the current C API strongly limits performance
- The ecosystem has to become more:
 - implementation agnostic
 - Python JIT friendly

We now know how we can get to this state
(while still supporting the current C API!)



A new API/ABI for faster Python: Python Native Interface

- Similar to HPy
- Implemented in CPython, meeting CPython needs
- Universal ABI: one wheel for all Python versions and implementations + debug mode without recompilation
- Python JIT friendly
- Usable by tools like Cython, nanobind, PyO3, Pythran, ...



Big project: needs funding and support

- Early state: a new PEP is being prepared
- Ecosystem-wide project
(CPython, Numpy, Cython, PyPy, GraalPy, etc.)
- How sufficient funding can be gathered?



Python Native Interface

Soon a presentation on this subject at PyConFR 2025

