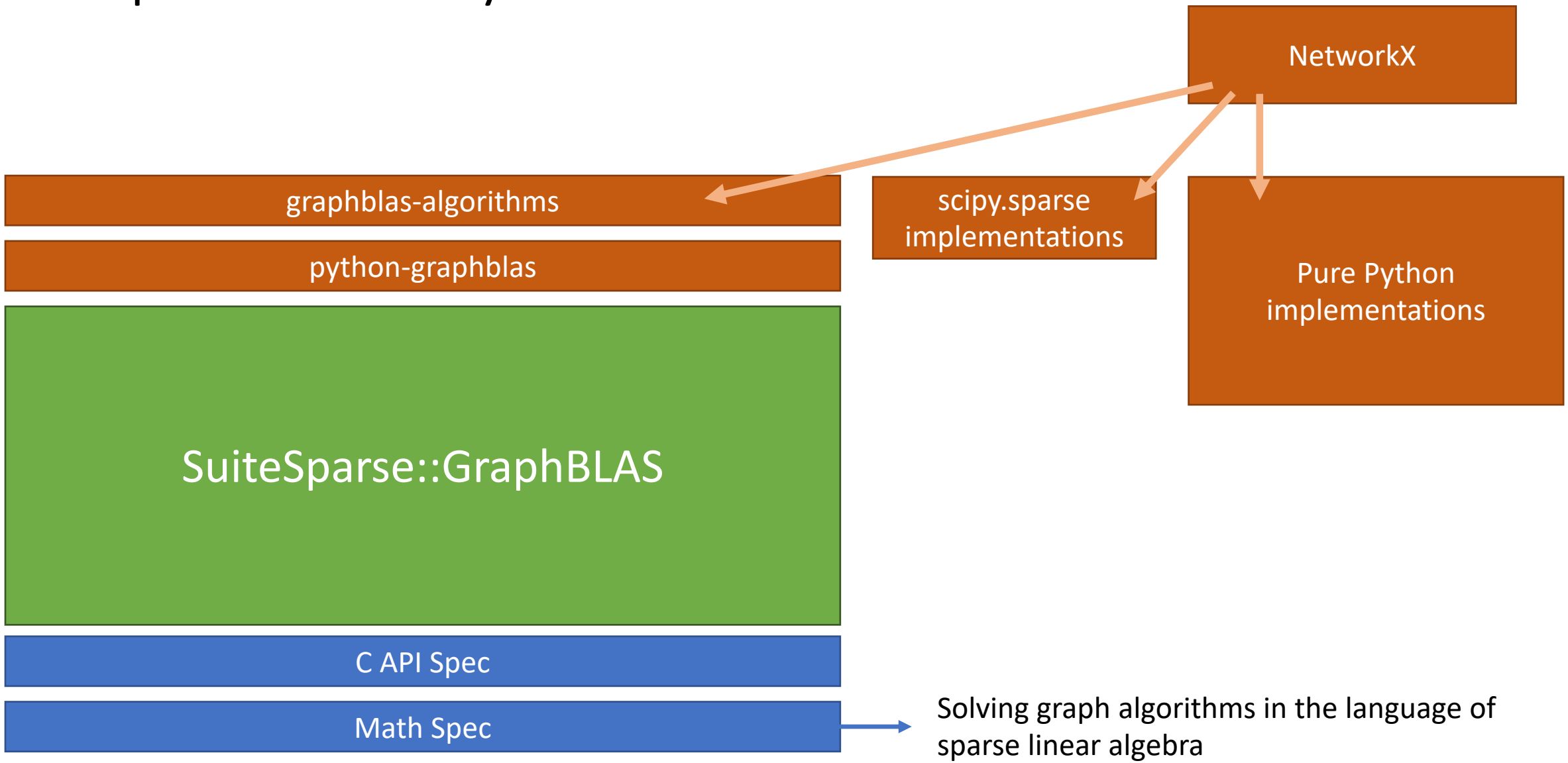
# GraphBLAS + Python Stack



# SuiteSparse::GraphBLAS

- missing ≠ 0
- Semirings → *plus_times* semiring $\sum_k a_{ik} * b_{kj}$ ; *min_plus* semiring $\min_k a_{ik} + b_{kj}$
- Masks → C(mask=M) << A @ B
- Multiple internal formats (CSR/CSC, DCSR/DCSC, Masked Dense)
- Highly tuned matrix multiply kernels (multi-core via OpenMP)
- Zero copy import/export of dense numpy arrays
- GPU support forthcoming
- No direct solvers

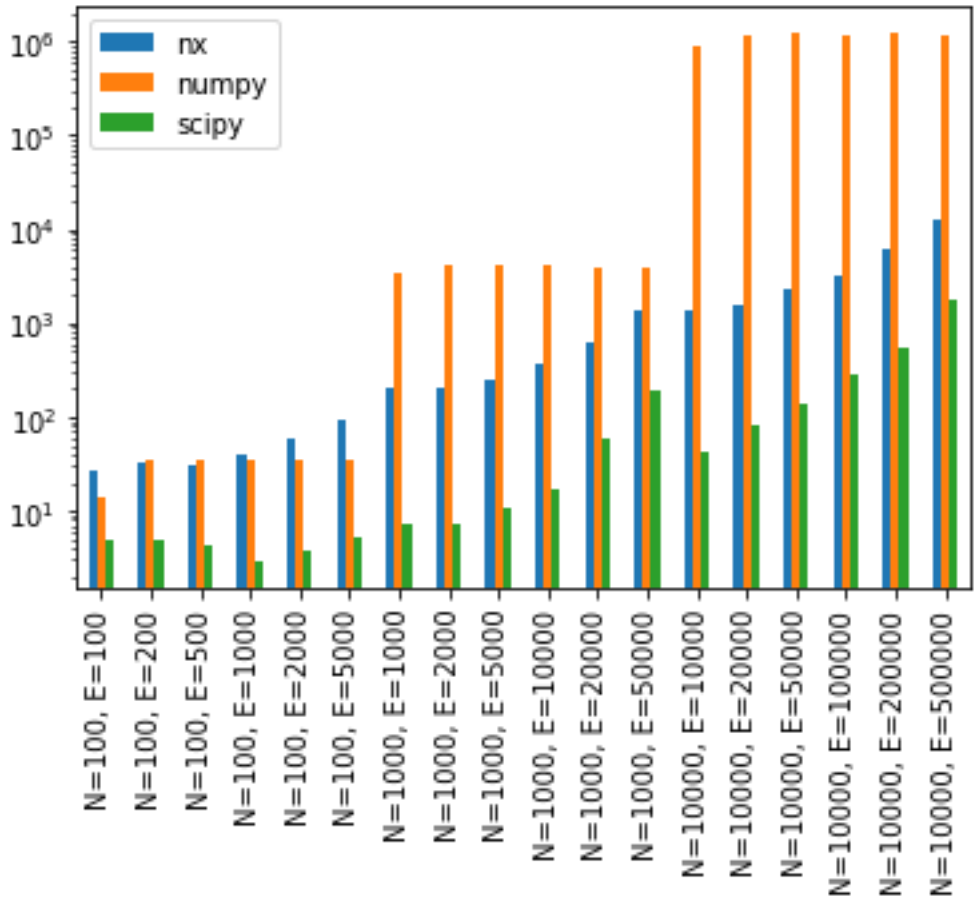
# Benchmarks

Speedup of Pagerank using GraphBLAS

| | # nodes | # edges | scipy | networkx |
|---|---|---|---|---|
| enron.txt | 36_692 | 367_662 | 2x | 48x |
| amazon.txt | 262_111 | 1_234_877 | 3.5x | 127x |
| google.txt | 916_428 | 5_105_039 | 12x | 281x |
| pokec.txt | 1_632_804 | 30_622_564 | 20x | 755x |

**Pagerank Performance**

nx: dict of dicts
numpy: dense matrix
scipy: sparse matrix




https://www.databulle.com/blog/code/python-pagerank-benchmark.html

# Collaboration

- Binary sparse storage format
  - Committee has been formed, working on v1 Proof-of-concept
  - Goal to be useful for all sparse libraries, not just GraphBLAS