

# JuliaSparse/SuperLUDIST.jl

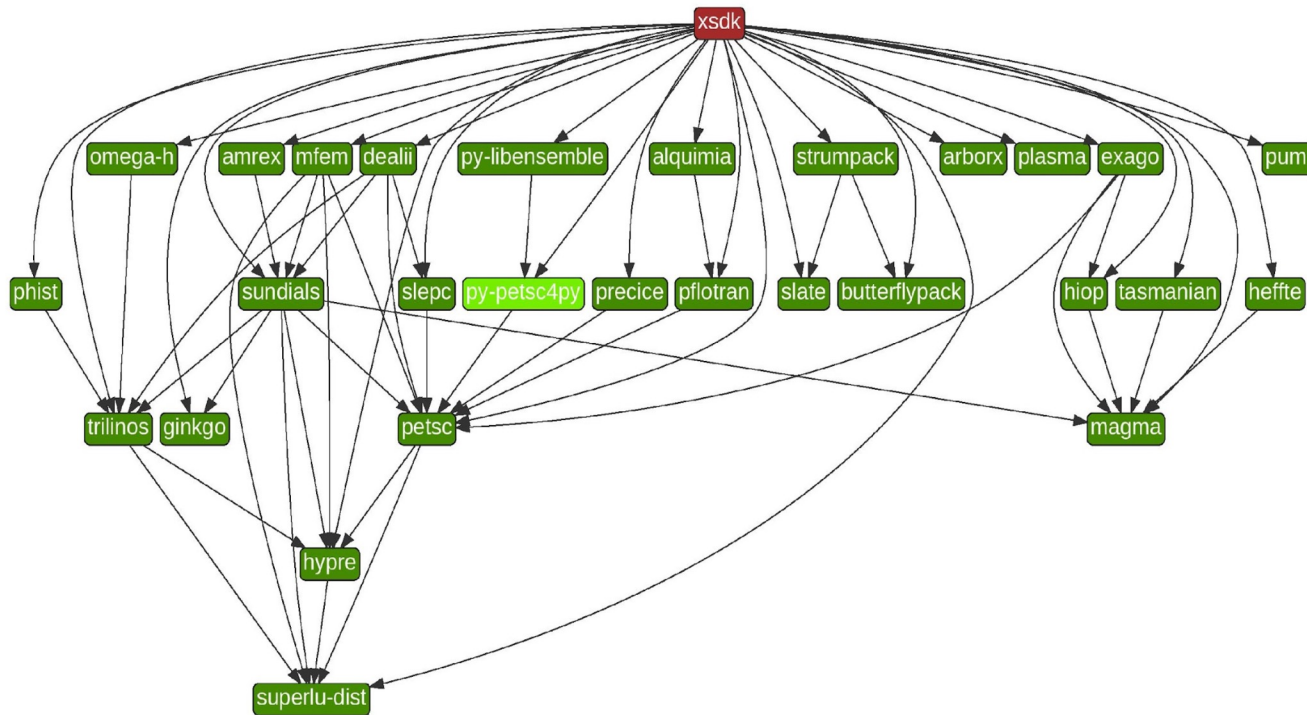
Example Julia interface for HPC math libraries

Sherry Li, Johannes Blaschke, Lawrence Berkeley National Laboratory

Raye Kimmrer, MIT

# ECP Math Libraries

- Common programming model: MP +X
- X = OpenMP, CUDA, HIP, SYCL, Kokkos, Raja
- xSDK helps installation, CI testing, interoperability



# MPI.jl: Julia bindings for MPI

- Simon Byrne, Lucas C. Wilcox, and Valentin Churavy (2021) "MPI.jl: Julia bindings for the Message Passing Interface". *JuliaCon Proceedings*, 1(1), 68, doi: [10.21105/jcon.00068](https://doi.org/10.21105/jcon.00068)
  - In Julia package registry since 2014
- Provide access to the existing MPI ecosystem of libraries
- NOTE: Julia's two other (native) modes of parallel computing
  - Multi-threading
  - Distributed computing

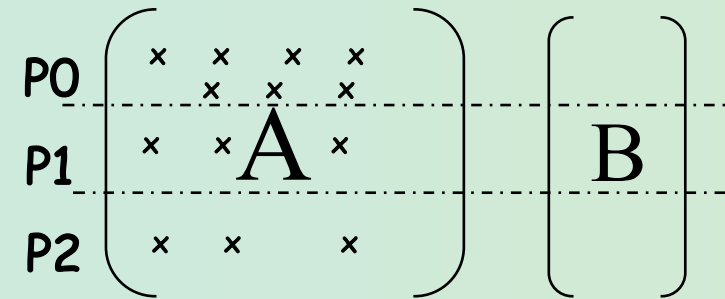
[ See Johannes' talk at workshop PAW-ATW, 11/13/23 ]

basic\_example.jl

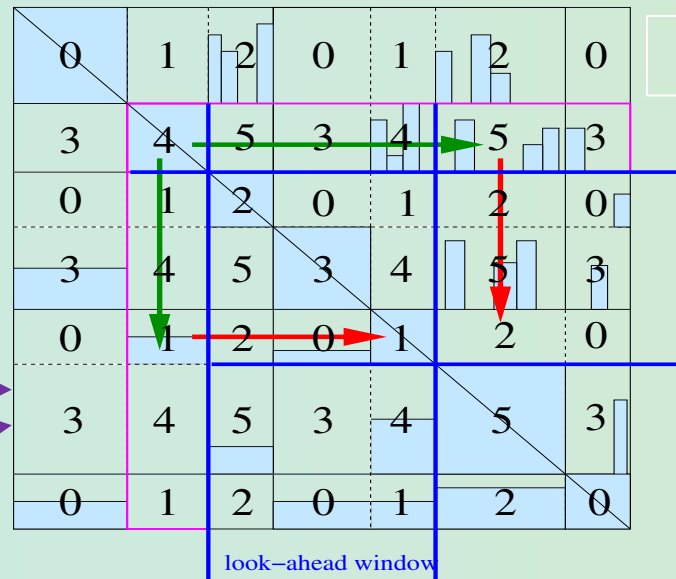
```
using MPI
using SuperLUDIST: Grid,
DistributedSuperMatrix, pgssvx!
```

```
MPI.Init()  
nprow, npcol, nrhs = 2, 3, 4  
comm = MPI.COMM_WORLD  
grid = Grid{Int64}(nprow, npcol,  
comm)
```

```
A = DistributedSuperMatrix{Float64,  
Int64}(grid), csr, chunksizes;  
root);  
b_local = # load RHS b  
...  
F = lu!(A);  
...  
x = F \ b_local
```



## 2D LU structures



## MPI Process Grid

0	1	2
3	4	5

# Maintain performance

Matrix torso3:  $n = 259,156$

LU factorization on Perlmutter at NERSC

	4 MPI	8 MPI	16 MPI
native	28.8	15.9	8.3
lu!(A)	32.3	18.4	10.5

Another example: <https://github.com/SciML/Sundials.jl>