Performance portability in PETSc

GPU

CPU C, C++, Fortran

Compiler Directives
OpenMP, OpenACC

Cuda PM CUDA, HIP **C++ Lambda PM**Kokkos, SYCL, RAJA

Application code

Using PETSc API

Front-end PETSc vector and matrix arrays are shared with user programming language/model

PETSc computation kernels

CPU: Use compiler options and vendor libraries for performance

GPU: Chosen for either speed of development or highest performance. Use GPU vendor libraries

C Code

BLAS/LAPACK MKL

OpenCL

OpenCL: ViennaCL

Back-end PETSc Vector and Matrix implementations

Cuda PM

CUDA: cuBLAS, cuSparse

HIP:rocBLAS, rocSPARSE

C++ data-parallel PM

Kokkos: Kokkos-kernels

SYCL: OneAPI