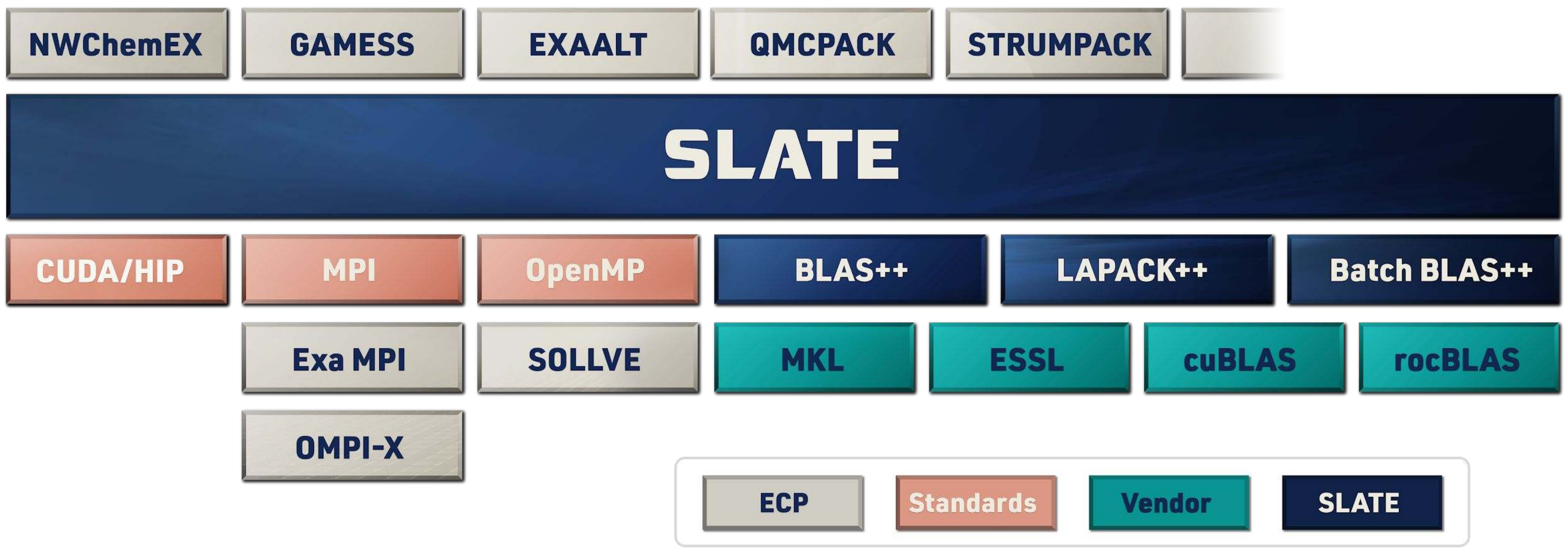
SLATE

SOFTWARE FOR LINEAR ALGEBRA TARGETING EXASCALE

SLATE will offer a modern replacement for ScaLAPACK.

SLATE will facilitate the development and advancement of multi-core and accelerator capabilities by leveraging recent progress and ongoing efforts in mainstream programming models (e.g., MPI 3+, OpenMP 4+, OpenACC). SLATE provides basic dense matrix operations (e.g., matrix multiplication, rank-k update, triangular solve), linear systems solvers, least square solvers, singular value and eigenvalue solvers.

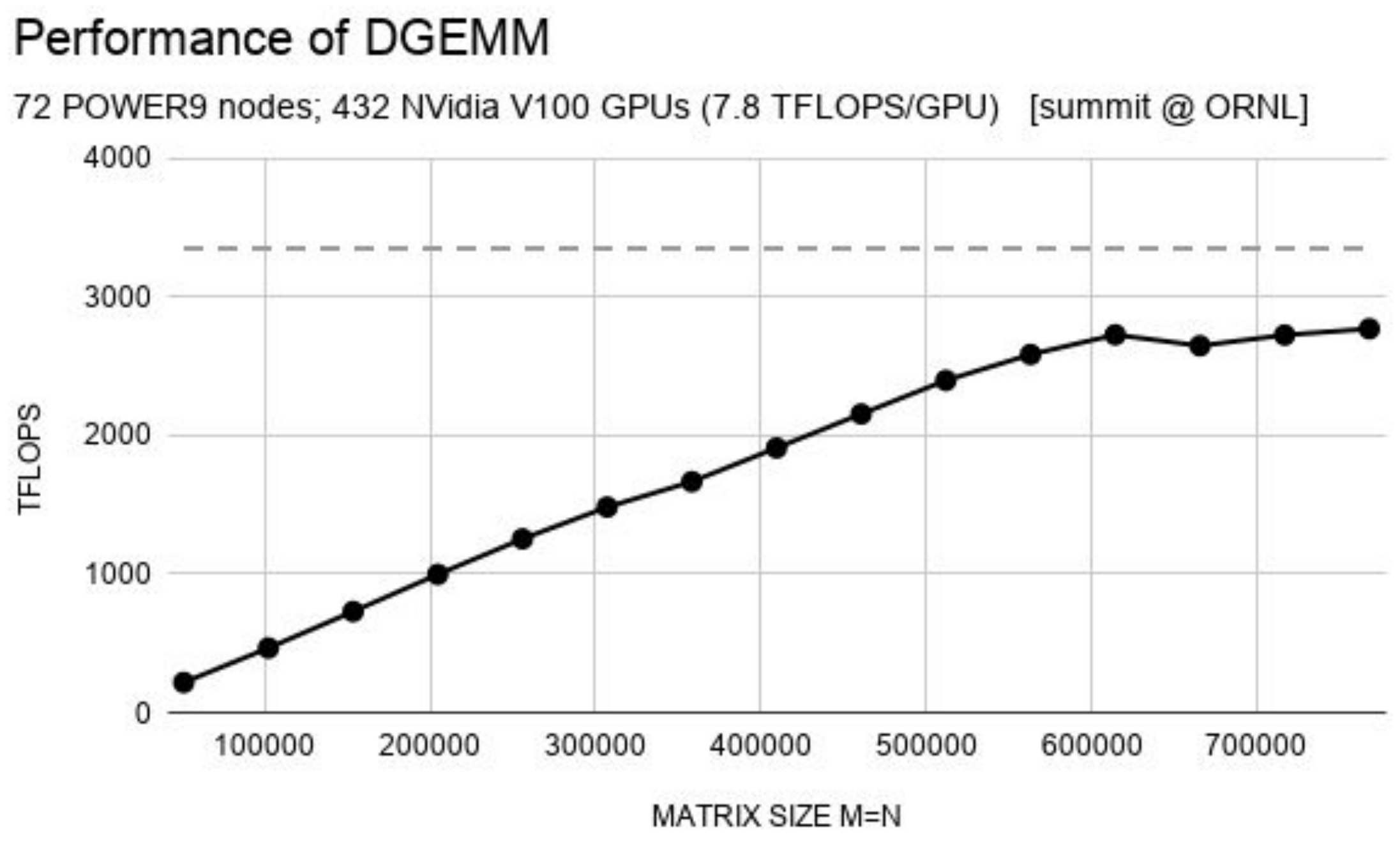
SLATE ARCHITECTURE



SLATE STATUS

PBLAS	GEMM, SYRK, SYR2K, HERK, HER2K, SYMM, HEMM, TRMM, TRSM
NORMS	Max, Frobenius, infinity, one norms for GE, TR, SY matrices
LINEAR SYSTEMS	Cholesky (LL^T), LU, Aasen's LTL^T
LEAST SQUARES	QR, LQ factorizations, least squares solvers
MATRIX INVERSIONS	Cholesky based inversion (POTRI), LU based inversion (GETRI)
SINGULAR VALUE, EIGENVALUE	Singular values (SVD), symmetric eigenvalues (SYEV)

PERFORMANCE



SLATE OBJECTIVES

COVERAGE	ScaLAPACK and beyond
HARDWARE	DOE CORAL (pre-exascale) → DOE Exascale
PORTABILITY	NVIDIA, AMD, Intel Xeon, IBM POWER, ARM; Standards: MPI + OpenMP + (Batch)BLAS
PERFORMANCE	Up to 80%–90% of peak (asymptotic)
SCALABILITY	Full Exascale machines Flexible data distributions, dynamic scheduling, overlapping communications
PRODUCTIVITY	~4 full-time developers
MAINTAINABILITY	Part-time developers + community

ScaLAPACK COMPATIBILITY API

Uses ScaLAPACK function names and signatures, i.e., no changes to the source code required (link time replacement). Environment variables are used to access SLATE specific functionality.

LAPACK COMPATIBILITY API

Uses LAPACK function signatures with a "slate_" prefix, e.g., slate_dgetrf(M, N, A, LDA, IPIV, INFO). There are additional settings through environment variables, e.g., "export LAPACK_NB=256."

BLAS++

https://bitbucket.org/icl/blaspp

Basic Linear Algebra Subprograms (BLAS) serve as the de facto standard for a performance-portable and numerically robust implementation of essential linear algebra functionality. BLAS++ provides a convenient, performance-oriented API for development in the C++ language and preserves established conventions while taking advantage of modern C++ features.

HIGHLIGHTS

- Covers the entire BLAS (~120 routines)
- Error handling with C++ exceptions
- Covered with a testing suite
- Documented with Doxygen

Mark Gates et al.
C++ API for BLAS and LAPACK
SLATE Working Note #2
http://www.icl.utk.edu/publications/swan-002

LAPACK++

https://bitbucket.org/icl/lapackpp

The Linear Algebra PACKage (LAPACK) is a standard software library for numerical linear algebra that provides routines for solving systems of linear equations and linear least squares problems, eigenvalue problems, and singular value decomposition problems. LAPACK++ provides a convenient, performance-oriented API for development in the C++ language and preserves established conventions while taking advantage of modern C++ features.

HIGHLIGHTS

- Covers majority of LAPACK (~1,200 routines)
- Error handling with C++ exceptions
- Covered with a testing suite
- Documented with Doxygen

Mark Gates et al.
C++ API for BLAS and LAPACK
SLATE Working Note #2
http://www.icl.utk.edu/publications/swan-002

BATCH BLAS++

http://icl.utk.edu/bblas

Many scientific and engineering computing applications solve large numbers of small and independent linear algebra problems. Such workloads can be executed much more efficiently on modern hardware if they are issued in large batches rather than one by one. To standardize the API, the HPC community is developing an extension to the BLAS standard called Batch BLAS. The objective of BBLAS++ is to provide a convenient, performance-oriented API for development in the C++ language that preserves established conventions while taking advantage of modern C++ features.

Ahmad Abdelfattah et al.
C++ API for Batch BLAS
SLATE Working Note #4
http://www.icl.utk.edu/publications/swan-004

ICL INNOVATIVE COMPUTING LABORATORY
THE UNIVERSITY OF TENNESSEE KNOXVILLE
SPONSORED BY
U.S. DEPARTMENT OF ENERGY Office of Science
ECP EXASCALE COMPUTING PROJECT
NNSA National Nuclear Security Administration

FIND OUT MORE AT
http://icl.utk.edu/slate/