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 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
- **Portability**: 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

**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

**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.

**HIGHLIGHTS**

- Batch BLAS++ provides a convenient, performance-oriented API for development in the C++ language.
- Supports large batches of independent linear algebra operations.
- Improves efficiency on modern hardware.

**Batch BLAS++ Further Reading**

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

---

**SLATE STATUS**

<table>
<thead>
<tr>
<th>BLAS++</th>
<th>LAPACK++</th>
</tr>
</thead>
<tbody>
<tr>
<td>GEMM, SYRK, SYR2K, HERRK, HER2K, SYMM, HEMM, TRMM, TRSM</td>
<td>Performance of DGEMM Performance of DGEMM</td>
</tr>
<tr>
<td>Norms: Max, Frobenius, infinity, one norms for GE, TR, SY matrices</td>
<td>72 PIONEER nodes, 432 NVIDIA V100 GPUs (7.5 TFLOPS/GPU) [Journal @ ORNL]</td>
</tr>
<tr>
<td>Linear Systems: Cholesky (LL^T), LU, Aasen’s LTL^T</td>
<td>1000</td>
</tr>
<tr>
<td>Least Squares: QR, LQ factorizations, least squares solvers</td>
<td></td>
</tr>
<tr>
<td>Matrix Inversions: Cholesky based inversion (POTRI), LU based inversion (GETRI)</td>
<td></td>
</tr>
<tr>
<td>Singular Value, Eigenvalue: Singular values (SVD), symmetric eigenvalues (SYEV)</td>
<td></td>
</tr>
</tbody>
</table>

**PERFORMANCE**

- **Performance of DGEMM**
  - 72 PIONEER nodes, 432 NVIDIA V100 GPUs (7.5 TFLOPS/GPU) [Journal @ ORNL]
  - 1000  |

---

**SLATE ARCHITECTURE**

**SLATE**

CUDA/HIP  | MPI  | OpenMP  | BLAS++  | LAPACK++  | Batch BLAS++ |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Programmable</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Acceleration</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HPC</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**SOFTWARE FOR LINEAR ALGEBRA TARGETING EXASCALE**

**SLATE**

Software for Linear Algebra Targeting Exascale

**SLATE ARCHITECTURE**

- CUDA/HIP
- MPI
- OpenMP
- BLAS++
- LAPACK++
- Batch BLAS++

**SLATE STATUS**

**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

**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

**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.

**HIGHLIGHTS**

- Batch BLAS++ provides a convenient, performance-oriented API for development in the C++ language.
- Supports large batches of independent linear algebra operations.
- Improves efficiency on modern hardware.

**Batch BLAS++ Further Reading**

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