xSDK4ECP

Extreme-Scale Scientific Software for ECP

The large number of software technologies being delivered to the application developers poses challenges, especially if the application needs to use more than one technology at the same time, such as using a linear solver from the PETSc/TAO mathematics library in conjunction with a time integrator from the SUNDIALS library. The xSDK project is an effort to create a value-added aggregation of mathematics and scientific libraries, to increase the combined usability, standardization, and interoperability of these libraries.

The ability to incorporate multiple libraries in a single executable is necessary as architectures become more complex and applications become reliant on multiple libraries to supply performant capabilities on those architectures to achieve their exascale performance and science goals. The Extreme-scale Scientific Software Development Kit (xSDK) is an effort to provide turnkey installation and use of popular scientific packages needed for next-generation scientific applications. The xSDK project is working to (1) enable the seamless combined use of diverse, independently developed numerical libraries as needed by exascale applications; (2) develop interoperability layers among numerical libraries in order to improve code quality, access, usability, interoperability, and sustainability; and (3) provide an aggregate build and install capability for the numerical libraries that supports hierarchical, modular installation.

The xSDK project focuses on community development and a commitment to combined success via quality improvement policies, better build infrastructure, and the ability the use numerical libraries in combination to solve large-scale multiphysics and multiscale problems. The project represents a different approach to coordinating library development and deployment. Prior to the xSDK, scientific software packages were cohesive with a single team effort but not across these efforts. The xSDK goes a step further by developing community policies followed by each independent library included in the xSDK. This policy-driven, coordinated approach enables independent development that still results in compatible and composable capabilities. Moreover, the xSDK provides a forum for collaborative numerical library development, helping independent teams to accelerate adoption of best practices, enabling interoperability of independently developed libraries, and improving developer productivity and sustainability of the libraries.

The xSDK project will also begin a coordinated effort to investigate and deploy multiprecision functionality in the ECP ST ecosystem to enable the use of low-precision hardware function units, reduce the pressure on memory and communication interfaces, and achieve improved performance. The project will assess current status and functionalities, advance the theoretical knowledge on multiprecision algorithms, design prototype implementations and multiprecision interoperability layers, deploy production-ready multiprecision algorithms in the xSDK math libraries, ensure multiprecision cross-library interoperability, and integrate multiprecision algorithms into ECP application projects.

PI: Ulrike Meier Yang, Lawrence Livermore National Laboratory

Collaborators: Lawrence Livermore National Laboratory, Argonne National Laboratory, Sandia National Laboratories, Lawrence Berkeley National Laboratory, University of California ¬– Berkeley, University of Tennessee – Knoxville, University of Oregon, NexGen Analytics, Oak Ridge National Laboratory, Karlsruhe Institute of Technology, University of Manchester, Charles University at Prague, Tech-X Corporation

Progress to date

- The xSDK team released version 0.4.0, which included 13 new xSDK members (AMRex, deal.II, DTK, MAGMA, MFEM, Omega_h, PHIST, PLASMA, PUMI, SLEPc, STRUMPACK, SUNDIALS, and Tasmanian) in addition to the original xSDK libraries (hypre, PETSc, SuperLU, and Trilinos), and the two domain components Alquimia and PFLOTRAN.

- The team continued development of the community policies. They refreshed the policies and added a new recommended policy including feedback from the ECP community. The policies were moved to github, and the process on changing or proposing policies has been updated.

- The team created reports on node-level resource management, which included survey results of efforts and future plans for the efficient transfer of resources of runtime library developers and approaches and plans of xSDK packages on their use of programming models and transfer of data resources; on the design, approach, and impact of the xSDK, summarizing history, community policies, release processes, library interoperability, xSDK usage, and impact on applications; and on the end-to-end use of the xSDK in three exascale applications.

- The team also interviewed many application teams on their needs of mathematical capabilities, computer usage, library usage, training, and more, to guide further xSDK development.