From *SIAM News*, Volume 38, Number 5, June 2005

CS&E 2005

Software Components in High-Performance Scientific Computation

By Lois Curfman McInnes and Lori Freitag Diachin

The Common Component Architecture (CCA) Forum (see www.cca-forum.org) began in 1998 as a grass-roots effort to bring about fundamental changes in the development and use of scientific software. The Forum is founded on the recognition of the burgeoning complexity and interdisciplinary nature of computational science and engineering, and on the vision that component-based software engineering (CBSE), which combines object-oriented design with the powerful features of well-defined interfaces, programming language interoperability, and dynamic composability, can help to overcome obstacles that make it difficult to share even well-designed traditional libraries.

This vision was not uncontroversial. Before the formation of the CCA Forum, there was little precedent in CBSE for some of the factors that are important in high-performance scientific computing, including parallelism in the single-program, multiple-data (SPMD) style, suitably efficient intercomponent communication, support for Fortran, support for a variety of high-performance architectures (including one-of-a-kind, leadership-class machines), and a tenable migration strategy for large legacy codes.

Accordingly, with funding from the U.S. Department of Energy's Scientific Discovery through Advanced Computing (SciDAC) initiative (see www.osti.gov/scidac), the CCA Forum established the Center for Component Technology for Terascale Simulation Software (CCTTSS, see www.cca-forum.org/ccttss), which is collaborating with several other SciDAC centers and applications groups to apply component concepts to high-performance computational science.

CBSE can be viewed as an evolutionary step beyond object-oriented design, with emphasis on the interoperability of software that has been independently developed by different groups. CBSE treats applications as assemblies of software *components* that interact with each other only through well-defined *interfaces* within a particular execution environment, or *framework*. Components are a logical means for encapsulating knowledge from one scientific domain for use by those in others. The glue that binds the components together are sets of domain-specific, community-defined "common" interfaces. Such common interfaces, as defined by working groups with expertise in various facets of scientific software, facilitate experimentation with different software tools that have similar functionality and alleviate the need for scientists to write separate code to manage interactions between an application and such tools.

The CCTTSS is a catalyst for interface-definition efforts in fusion, climate, chemistry, and combustion, as well as in numerical tools under development at the SciDAC Terascale Simulation Tools and Technologies (TSTT) Center (see www.tstt-scidac.org), which focuses on interoperable meshing and discretization technologies, and at the SciDAC Terascale Optimal PDEs (TOPS) Center (see www.tops-scidac.org), whose mission is to develop a set of compatible toolkits of open-source, optimal complexity solvers for nonlinear PDEs. For example, the TSTT Center is developing interfaces for managing mesh, geometry, and field data for the numerical solution of PDEs, and the TOPS Center is creating a prototype set of linear algebra interfaces. Both the TSTT and TOPS interfaces use the Scientific Interface Definition Language (SIDL), a programming-language-independent specification that facilitates interaction among software developed in different programming languages, such as Fortran, C, C++, and Python; SIDL gives users access to a wide variety of existing libraries, as well as new implementations.

The accompanying figures illustrate several facets of joint work under way at SciDAC-funded centers. Unstructured meshes are employed in many PDE-based models, including the design of the next generation of particle accelerators by the Stanford Linear Accelerator Center (SLAC). Figure 1A shows a typical mesh used in the design of the accelerating

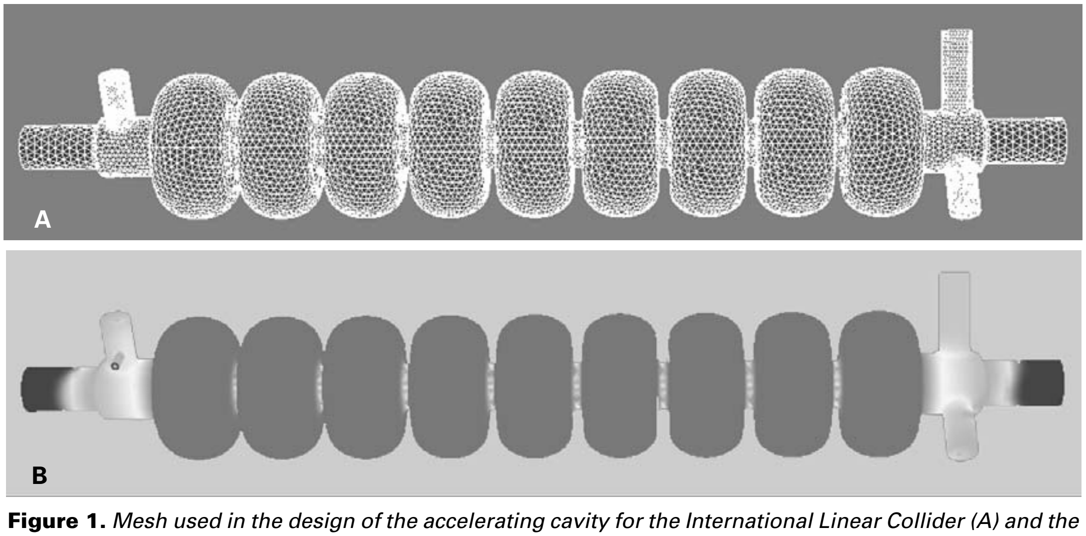

Figure 1. *Mesh used in the design of the accelerating cavity for the International Linear Collider (A) and the resulting computed electric field distribution of the accelerating mode (B). Courtesy of the Advanced Computations Department, SLAC.*

cavity for the International Linear Collider (ILC); the resulting electric field distribution of the computed accelerating mode is shown in Figure 1B. In the overall design process, several hundred additional dipole modes in such a cavity need to be computed as the end-cell geometries are optimized to minimize the effect of these modes on the particle beam.

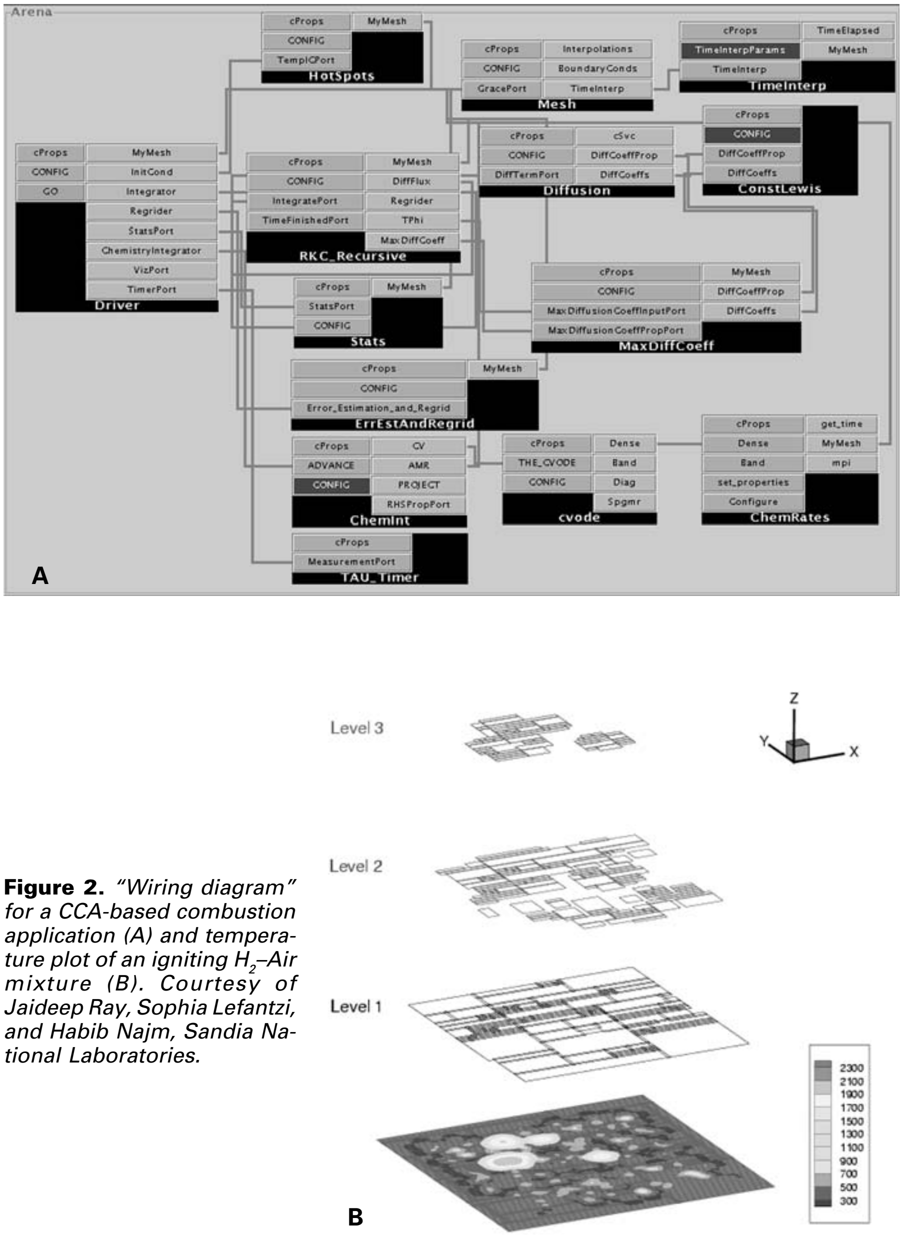

Figure 2. *"Wiring diagram" for a CCA-based combustion application (A) and temperature plot of an igniting H_2–Air mixture (B). Courtesy of Jaideep Ray, Sophia Lefantzi, and Habib Najm, Sandia National Laboratories.*

Through a project called "Advanced Computing for 21st Century Accelerator Science and Technology" (see scidac.nersc.gov/accelerator), SLAC is collaborating with the TSTT and TOPS Centers to make this design-optimization computation possible. The TSTT Center is developing the mesh- and geometry-modification components needed for the simulation, including mesh-quality improvement and adaptive mesh refinement, and TOPS is providing the necessary optimization and sensitivity-analysis tools.

In another example, researchers in the CCTTSS and the Computational Facility for Reacting Flow Science (CFRFS, see cfrfs.ca.sandia.gov) are working together on a component-based toolkit for flame simulations. In collaboration with the Performance Evaluation Research Center (PERC, see perc.nersc.gov), the Algorithmic and Software Framework for Applied Partial Differential Equations Center (APDEC, see davis.lbl.gov/apdec), and the TOPS Center, this group has developed components for time integration, structured adaptively refined meshes, performance observation, and various chemical and physical models.

By making the incorporation of numerical expertise straightforward, the component-based architecture has allowed the group to explore the use of high-order discretization on adaptively refined meshes. Figure 2A shows a "wiring diagram" for a CCA-based combustion application; the black boxes represent various components. While at this resolution the image does not convey the details of component interactions, it does make the key point that interactions between components take place only via well-defined interfaces, or ports; in the diagram, ports provided and used by components are denoted by dark and light shaded boxes, respectively. Figure 2B shows a temperature plot of an H_2–Air mixture ignited by random temperature kernels. The temperature is plotted on the base (100-μm) mesh. The diagrams above demarcate the regions of refinement, with each level of refinement corresponding to a factor of two.

These examples are provided to give readers a glimpse of ongoing work in high-performance scientific components. Research on a wealth of related issues in applied mathematics and computer science is under way. Further details can be found at www.cca-forum.org.

Lois Curfman McInnes is a researcher in the Mathematics and Computer Science Division of Argonne National Laboratory, where her research focuses on algorithms and software for large-scale nonlinear partial differential equations. She is a founding member of the CCA Forum and the leader of the Scientific Components thrust of the CCTTSS (one of four primary research areas of the center). Lori Freitag Diachin is a member of the Center for Applied Scientific Computing at Lawrence Livermore National Laboratory, where her research focuses on mesh-quality improvement and interoperable meshing and discretization component development. She is a principal investigator of the TSTT SciDAC Center.