Grid computing and Web services

The next time you take a plane, thank a multi-disciplinary team led by CS professors Keshav Pingali and Steve Vavasis and Civil & Environmental Engineering professor Tony Ingraffea for making your flight a little safer.

Most commercial and military aircraft are flown well past their intended lifetime —the B-52 will be flown until 2040 when it will be 94 years old, three times its original planned lifetime. The most critical problem with aging aircraft is wear and tear, which causes microscopic cracks to form in the fuselage and engines, compromising airworthiness.

To simulate how cracks form and propagate in mechanical structures like airframes and rocket engines, the team is using grid computing. "The grid-computing metaphor represents many styles of distributed computing," said Pingali. "We have shown that some of the more useful styles of grid computing can be done quite effectively using existing industry-standard protocols and software such as SOAP and XML."

The benefits of using industry-standard Web services became evident while the team was exploring a simulation of fracture in engine components. These components transport high-pressure, high-velocity chemically reacting gases, which can create large thermo-mechanical stresses on component walls. To simulate fracture initiation and growth, the group had to integrate several software systems, including a finite-element mesh generation code developed jointly by Cornell and the College of William and Mary, a chemically reacting flow-simulation code developed at Mississippi State University (MSU) and the University of Alabama, and a linear elastic fracture code developed at the Cornell Theory Center.

> This is one of the few projects that I would call a Web service, and it is well beyond what is running on today's experimental grid.
>
> Gordon Bell
> Microsoft Research

"The traditional approach to integrating such software modules is to port them all to a single computing platform," said Pingali. "Not only is this time-consuming, but every time a new release of a module becomes available, some poor soul has to repeat the entire process of downloading and porting the code, recompiling it, relinking the compiled code with the rest of the software, and so on."

To simplify the job of integrating software components while respecting individual software developers' choices of hardware platforms, operating systems, and programming languages, the team decided to deploy each major component as a Web service running on a server at the institution where the component was developed. The flow-simulation code, for example, runs on an IBM x330 Linux server at MSU, while the fracture simulation code runs on the Cornell Theory Center's Windows cluster. Industry-standard Web service implementations, such as Apache SOAP and XML-based data exchange formats developed by Vavasis, are used.

"We view the person running the simulation as a client who writes a few hundred lines of code to invoke the various Web services to orchestrate the simulation," said Vavasis. "Our motto is 'write once, run from anywhere.'" He said that the overhead of using geographically-distributed Web services for their simulation is about 10 percent.

Gordon Bell, senior researcher at Microsoft's Bay Area Research Center, said, "This project demonstrates the potential for a new way to build applications and the potential for a new software industry structure based on delivering results. Users don't have to buy apps programs and maintain a more complex software environment; they simply call a program or database. This is one of the few projects that I would call a Web service, and it is well beyond what is running on today's experimental grid."

The team's next project, being done in collaboration with NASA, is to develop an adaptive control system to protect civilian airliners from attack by shoulder-fired missiles. "A network of sensors and processors embedded in the aircraft structure provides a nervous system for sensing and estimating damage," said Pingali. "Given this information, the system must perform projections of aircraft performance and adjust flight controls appropriately."

Structural failure in Aloha Airlines Boeing 737 (28 April 1988).

Keshav Pingali (center), Radu Rugina (left), and Steve Vavasis use grid computing to study fracture in mechanical structures like the portion of a Boeing 747 fuselage that Pingali is holding.

Another ongoing project is focused on providing transparent checkpoint-restart capability for long-running scientific programs in computational grids, which will enable applications to migrate between sites to take advantage of changing resource availability. CS professor Radu Rugina is working with CS senior research associate Paul Stodghill on approaches to reduce the amount of state saved at these checkpoints. This work builds on Rugina's successes in developing sophisticated compiler analyses and transformations to reduce the memory requirements of Java programs and to automatically find memory errors in C programs.

CS chair Charlie Van Loan believes that these projects exemplify the interdisciplinary research style made possible by the collegial atmosphere at Cornell. "Like many schools, we have a strong computational science and engineering program," said Van Loan, adding, "but what is unique here is the collaboration among numerical analysts like Vavasis, practitioners like Pingali and Rugina, and faculty in engineering disciplines like Ingraffea. I cannot think of many other places in the world where this kind of work can be done."

The Cornell Theory Center

The Center for Theory and Simulation in Science and Engineering, or Cornell Theory Center (CTC) as it is known unofficially, was founded in 1984 as one of five national supercomputing centers funded by the NSF to provide high-performance computing to researchers across the country. Ken Wilson, the 1982 Nobel laureate in Physics, was its founding director.

Large-scale delivery of cycles to researchers started on the IBM 3090 series of vector computers. A few years later, CTC deployed one of the largest IBM SP parallel computers in the world. More recently, CTC has pioneered the use of Windows clusters for high-performance scientific computing, with over 2000 processors. Substantial funding for this effort was provided by Microsoft, Dell, and Intel. The results from this project were highly influential in persuading Microsoft to enter the high-performance computing market. CTC is now an important unit of the Faculty of Computing and Information Sciences.

From its inception, the CTC has had strong ties with CS. The original proposal to the NSF to be one of four national supercomputing centers included contributions from CS professors John Hopcroft, Ken Birman, Bob Constable, and Charlie Van Loan.

As CTC's mission evolved under NSF's ten-year supercomputing centers program, the relationship with CS was formalized through establishment of the Advanced Computing Research Institute (ACRI), led by CS professor Tom Coleman. The ACRI's mandate was to apply CS expertise in parallel linear algebra methods, advanced optimization techniques and applications, parallel solution of PDEs, and the study of parallelizing compilers for scientific computing to a range of scientific applications. ACRI members Keshav Pingali, Steve Vavasis, and Charlie Van Loan collaborated with groups in biomechanics, Chemistry, and Civil & Environmental Engineering. The ACRI used one of the first Intel Hypercubes and solidified CTC's reputation among computational scientists and engineers as a pioneer in parallel computing.

Tom Coleman became director of CTC in 1997 and created even stronger CS linkages. During his tenure, Pingali, Ron Elber, and Johannes Gehrke were CTC Associate Directors. Pingali forged a strong alliance with Cornell professor Tony Ingraffea to create the Computational Materials Institute (CMI), which successfully competed for large-scale proposals from NSF's Knowledge and Distributed Intelligence and Information Technology Research programs. Elber led the center's NIH-funded Parallel Processing Resource for Biomedical Scientists and continues to lead CTC's Computational Biology Service Unit; both initiatives have contributed to Cornell's reputation as a powerhouse in computational life sciences.

A major new thrust is data-intensive computing. Led by Gehrke, this project is extending the expertise of the CS database group to an increasingly broad base of collaborators in fields from proteomics to astronomy.

Given this rich history of interdisciplinary work, it is not surprising that CTC plays a central role in the CIS mission to transform intellectual disciplines across the university by making them more computational.

Cornell's Rhodes Hall houses the Theory Center.

1990

With completion of Rhodes Hall, CS expands to 38,000 sq. ft. of space.

David Gries receives the ACM SIGCSE Award for Contributions to CS Education.

David Gries receives the CRA (Computing Research Association) Award for Service to the CS Community.

Juris Hartmanis is elected a Foreign Member of the Academy of Science of Latvia.

John Hopcroft receives an honorary doctorate from the University of Seattle in Washington.

Charlie Van Loan becomes a member of the SIAM Council.

Tom Coleman and Yuying Li publish *Large-scale Numerical Optimization* (SIAM Publications).

Paul Pedersen, Carlo Tomasi, Nick Trefethen join.

1991

The CS research budget tops $6 million. CS receives an NSF grant on Revitalizing the Computer Science Curriculum and acquires an 8000-node CM-200 data parallel computer.

Don Greenberg is elected to the National Academy of Engineering.

John Hopcroft is appointed to the National Science Board, which oversees the National Science Foundation.

John Hopcroft becomes Chair of the Board of Trustees of SIAM.

Dexter Kozen publishes *The Design and Analysis of Algorithms* (Springer-Verlag).

Steve Vavasis publishes *Nonlinear Optimization: Complexity Issues* (Oxford Science).

Tom Henzinger, Ronitt Rubinfeld join. Juris Hartmanis becomes Chair. John Hopcroft becomes Dean of Engineering.

1992

Dick Conway is elected to the National Academy of Engineering.

Dexter Kozen receives a Prize from the Polish Ministry of Education.

Students Aravind Srinivasan and Alessandro Panconesi receive the Best Student Paper Award at the ACM Symposium on the Theory of Computing.