



Clockwise from left: Laxmikant Kalé, Zan Luthey-Schulten, Emad Tajkhorshid, Aleksei Aksimentiev, Klaus Schulten

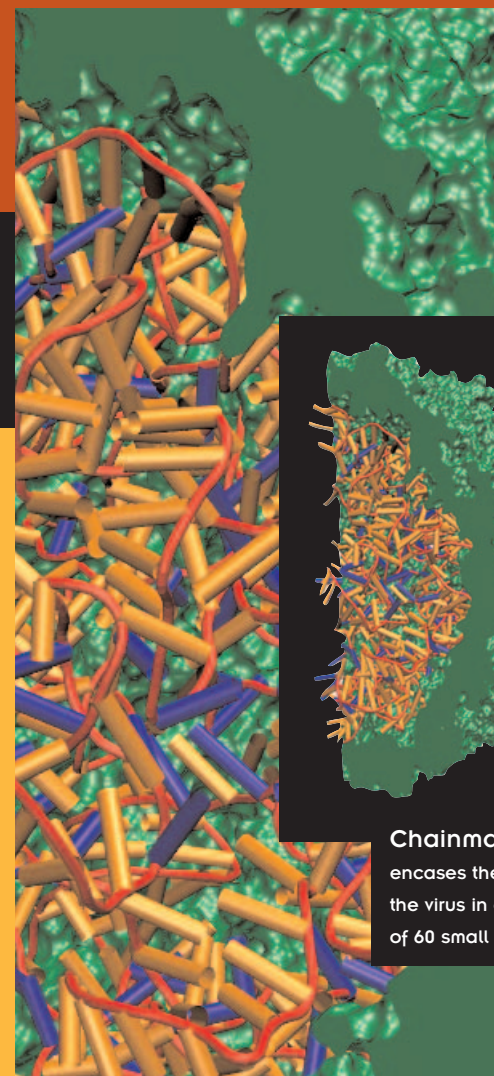
The TCB group includes five U of I faculty – computer science professor Laxmikant Kalé, chemistry professor Zan Luthey-Schulten, biophysics assistant professor Emad Tajkhorshid, physics assistant professor Aleksei Aksimentiev, and physics professor Klaus Schulten, the group's director.

The TCBG is known internationally for their large scale simulations of biomolecules, molecular dynamics programs, visualization software, collaboration software and more. TCBG software is used by tens of thousands of scientists worldwide.

TCBG

Theoretical and
Computational
Biophysics Group

Bringing
Computing
to Life



Chainmail. The STMV capsid encases the genetic material of the virus in an interlocking mesh of 60 small proteins.

Theoretical & Computational Biophysics
Beckman Institute for Advanced Science & Technology
University of Illinois at Urbana-Champaign
405 N. Mathews Ave.
Urbana, IL 61801
<http://www.ks.uiuc.edu>

Beckman Institute for Advanced Science and Technology
University of Illinois at Urbana-Champaign

John Stone had always liked a challenge. And he'd written software for almost every personal computer available. So when a fellow graduate student enlisted him to speed up software that predicted how air would flow through jet engines, Stone jumped at the chance.

Over the next several weeks, Stone, then a master's student at the University of Missouri-Rolla, merged his high-end graphics program with his friend's fluid-simulation program. His friend had been spending two and half days of computer time to create a single snapshot of turbulent air flow. The merged program produced the same snapshot in seven seconds, and soon the two had created an animation of turbulent airflow. "He could see something he couldn't see before," Stone recalls. "It was great for both of us."

A decade later, Stone is still helping scientists uncover nature's mysteries. As a senior research programmer at Beckman Institute's Theoretical and Computational Biophysics (TCB) group, he develops state-of-the-art software that helps scientists visualize the movement of large biological molecules. And he's got plenty of company. The TCB group includes a six-member team of developers who create parallel-computing software that simulates biomolecules, visualizes the results of those simulations, and develops new software for scientific collaborations.

Their software is used by tens of thousands of scientists worldwide, it's won them a prestigious Gordon Bell award for exemplary use of a large parallel computer, and it's helped make the TCB group one of the world's top biophysics laboratories.

And key to their success are computer scientists like Stone. By delving deep into a few scientific applications, they've enabled scientists to probe the far edge of biophysics, and they've developed some of the most powerful parallel-computing software on earth.

A High-tech Haven

A unique hybrid of computer science and computational science, the TCB group excels in both. They operate the National Institutes of Health Resource for Macromolecular Modeling and Bioinformatics, which develops new algorithms and computing tools to shed light on the molecular machines that cells use to harvest energy from sunlight, sense their environment, and turn on genes.

With more than \$2 million a year in external funding, the TCB group includes five U of I faculty – physics assistant professor Alek Aksimentiev, computer science professor



Jim Phillips (above) and John Stone (right), Senior Research Programmers in the TCBG, develop state-of-the-art software.

TCBG developers have a rare opportunity to improve their software by working directly with its end users. It's an opportunity many commercial programmers never get, and one that Stone and Phillips sought out. Today, Stone develops and maintains TCBG's Visual Molecular Dynamics (VMD) program, and Phillips NAMD, a scalable molecular dynamics program.

Laxmikant Kalé, chemistry professor Zan Luthey-Schulten, biophysics assistant professor Emad Tajkhorshid, and physics professor Klaus Schulten, the group's director. They're known internationally for their huge simulations of biomolecules and for their software, including molecular dynamics programs, visualization software, collaboration software and more.

Since the early 1990s, TCB computer scientists have pioneered the scientific use of parallel computing clusters, with each cluster giving way to one more powerful. The three latest clusters, named after moons of Uranus, each consist of 24 dual-processor, rack-mounted Xeon systems, all housed in the group's Beckman Institute facilities. The TCB group also runs its software on some of the country's most powerful supercomputers, including the 10-teraflop Xeon Linux cluster at the nearby National Center for Supercomputing Applications.

But it's not just high-end technology that has made the TCB group a haven for computer scientists. The group's developers immerse themselves in a single project for several years at a time, and they're completely involved in all the decisions about the software. Such autonomy is rare in the commercial world, says senior research programmer Kirby Vandivort, who's been working with Schulten since 1999. "I think a lot of commercial programmers are told, 'Make this widget do that.' That's not nearly as exciting as having to go through the design process in your head and figuring out what you want to do."

And these days, what TCB group developers often want to do is make scientific software faster.

The Need for Speed

From the beginning, the TCB group pursued the fastest parallel computers they could find. In 1988, soon after arriving at the Beckman Institute, Schulten realized that to understand the key reactions of photosynthesis, a process that ultimately supplies the energy used by most of the planet's creatures, he had to create a realistic computer model of a key photosynthetic protein and the cell membrane it sat in. That meant simulating the behavior of 27,000 atoms for a nanosecond – a huge molecular-dynamics simulation that could then be done only on a Cray supercomputer that cost more than \$10 million.

“Nobody was going to give me that computer. So we built this one.” Schulten points to a modest stainless steel box on

a shelf in his office. It's a 60-processor parallel computer, and Schulten and his graduate students spent months building it, soldering their own circuit boards, rehabbing an oscilloscope to test circuits, and printing circuit board masks with an aging laser printer. To simulate the behavior of the 27,000-atom protein, they developed a novel computer program and ran the computer continuously for more than two years. It was one of the earliest uses of parallel computing in the life sciences.

Schulten's team published the results in 1993, and soon began running their molecular-dynamics simulations on the first commercial parallel computers, including the massively parallel Connection Machine. The TCB group's physicists and chemists always wrote their own software.

Eventually, in collaboration with computer scientists, they created a program called NAMD to perform molecular-dynamics simulations on parallel machines, using a then-new programming language called C++ that made the program more modular and more extendable.

But they needed to compute even faster if they were to understand the behavior of important biomolecules containing more than 100,000 atoms. And as parallel-processing clusters grew larger, NAMD ran slower instead of faster. It became clear that the real bottleneck was in the software.

A Two-Way Street

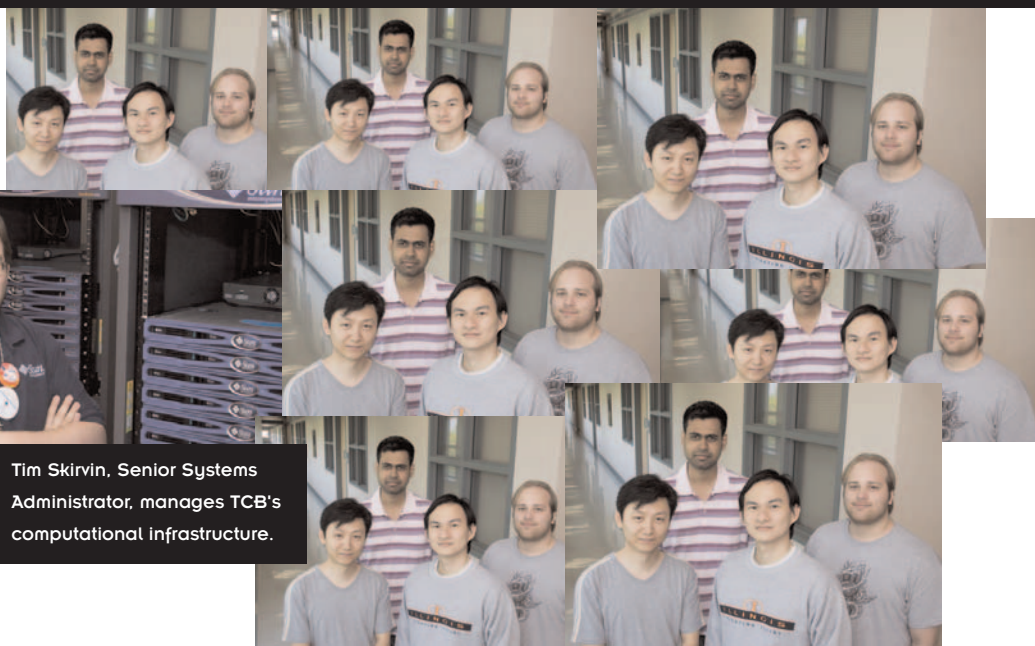
To come up with a solution, Schulten early on enlisted the help of Kalé, a computer-science professor who currently heads the University of Illinois' Parallel Programming Laboratory. Kalé and several of his grad students got to work immediately. To make NAMD calculate as fast as the scientists wanted, they needed a very efficient way to make use of all the processors in a cluster. Their answer was a parallel programming system called Charm++ that conducts automated resource management, basically acting like a factory-floor supervisor to steer work to idle processors.

Jim Phillips, a physics graduate student in the TCB group, then made sure those processors did their job correctly. To calculate the movement of the biomolecules with thousands of atoms, Phillips wrote algorithms to calculate how each pair of atoms in the protein tugs or pushes each other, and how these forces combine to move the entire protein. Phillips' work ensured that “when we run simulations on 100 or 1,000 processors, we're getting the right answers,” he says.

Today, NAMD can simulate the movement of proteins with millions of atoms, making it the world's fastest parallel molecular dynamics program. Phillips, currently a senior research programmer in the TCB group, continues to incorporate the latest parallel-computing advances into NAMD, which now runs efficiently on several thousand parallel processors.

Working on scientific software offers a different type of challenge to mathematically inclined computer scientists. Robert Skeel, a longtime UI computer-science professor and TCB group member who recently moved to Purdue University, develops numerical

Schulten early on enlisted the help of Kalé, a computer science professor who currently heads the University of Illinois' Parallel Programming Laboratory. Kalé and several of his grad students got to work immediately. Kalé students Gengbin Zheng, Sameer Kumar, Chee-Wai Lee, and David Kunzman are shown below.



Tim Skirvin, Senior Systems Administrator, manages TCB's computational infrastructure.

Charm++ now undergirds cutting-edge scientific modeling software being developed by Scientists for fields as diverse as materials science, quantum chemistry, rocket engineering, and astronomy.

What makes the collaboration ideal from a computer scientist's point of view, Kalé says, is that it's a "two-way street." Computer scientists are more than just programmers fulfilling the directives of scientists, as sometimes happens. Instead, they've been challenged to develop "an enabling technology that applies to a broad variety of problems."

methods that enable NAMD to more efficiently calculate the forces that charged atoms exert on the atoms around them, which in turn enables NAMD to run faster and simulate the behavior of ever-larger biomolecules. "It's satisfying to program things and see if the mathematical theory works, especially when the mathematical theory tells you things you might not otherwise see," Skeel says.

TCB developers also have a rare opportunity to improve their software by working directly with its end users. It's an opportunity many commercial programmers never get, and one that John Stone has sought out since his early collaboration on airflow in jet engines. Today, Stone develops and maintains the TCB group's Visual Molecular Dynamics (VMD) program, which is used by 40,000 scientists worldwide to translate the output of NAMD into a visual display that shows how biomolecules move. He also fields dozens of emails each day, providing technical help to VMD's users, who in turn request a never-ending set of new features. As the senior research programmer in charge of VMD, Stone decides which features to implement, when, and how.

Such synergy is also a key component of BioCoRE, TCB-group software that allows biologists to work together from remote locations. BioCoRE's users can monitor remote jobs running on supercomputers, run visualization software like VMD, chat about a project, or update a lab book. Some of those users – the TCB group's biophysicists – are right down the hall. Vandivort jokes about the "feedback" he gets from users, who never cease telling him how to improve the program, but adds that "it's nice to see the fruits of your labor being used."

The Joy of Computing

More than a decade after Schulten and the TCB group built their first parallel cluster, they rely on parallel computing more than ever – a situation that's now common in a variety of scientific fields. But many challenges remain. TCB developers are adapting



It's not just high-end technology that has made the TCBG a haven for computer scientists. Developers immerse themselves in a single project for several years at a time, and are completely involved in all the decisions about the software. Such autonomy is rare in the commercial world, says senior research programmer Kirby Vandivort.

NAMD to run on the largest parallel machines in the country, such as IBM's over 131,000-processor Blue Gene, soon to be installed at Lawrence Livermore National Laboratory. NAMD also needs better performance analysis, performance visualization, dynamic load balancing, plus it should all be wireless and portable, Kalé says. Scientists "should be able to check on the progress of a simulation in an airport from their PDA," Kalé says.

Such improvements in NAMD and other TCB software will help biologists probe the very stuff of life. But pushing parallel computing does more, Kalé says. With scientific fields as diverse as climate science, economics,

and drug design now relying on massive computations for new insights, advancing high-end parallel computing will help the U.S. maintain its scientific and technological edge, contributing to the "broader good of society," Kalé says.

The TCBG developers share in Kalé's perspective. Working on cutting edge projects, on state of the art hardware, and interacting directly with users, are compelling attributes of their work for developers. But a more fundamental reward is knowing their software contributes to the work of others. Says Vandivort: "We solve problems that enable others to find solutions."