The Cornell CS ambience

Computer Science at Cornell opened its doors in 1965, with just an MS-PhD program. The field's first task was to produce faculty to populate the future CS departments.

Long before computer science became a mature discipline whose technology is changing the face of the world, the Cornell CS Department believed that computer science was a deeply coherent intellectual discipline. We saw its traditional scientific character, the interplay between theory and experiment, and we imagined traits never seen before, such as the ability to design in ways nearly unconstrained by the physical world.

Juris Hartmanis, the founding chair and a Turing Award winner, who helped shape the theoretical character of the discipline, gave us a distinguished presence in computing theory. John Hopcroft joined in 1967 and soon became a leading theoretician, eventually winning the Turing Award for his fundamental contributions to the field of algorithms.

Founding member Gerry Salton, the father of information retrieval, helped establish the experimental side of computer science. Gerry's experiments with his SMART system gave rise to the vector space model and other technical concepts on which today's search engines are built and which helped create the new cyberspace frontier. More recently, Jon Kleinberg's fundamental work on ranking Web documents illustrated a similarly powerful synergy between theory and practice. His work, which is based on an elegant hubs-and-authorities model using fixed points in high-dimensional vector spaces, has greatly influenced how search engines rank pages.

The best theory is inspired by practice. The best practice is inspired by theory.
Donald E. Knuth

The renowned Program in Computer Graphics (PCG) was founded in 1973 by Don Greenberg. Its theoretical work on light reflection models and surface modeling was directed toward synthesizing realistic images. The PCG succeeded, with several *Scientific American* covers, five SIGGRAPH achievement awards, and five alumni with Hollywood's technical Oscars, including CS professor Steve Marschner.

We came to stress the interplay between theory and practice in system design. For instance, many concepts and components in Ken Birman's Isis system, which is still in use in the NY Stock Exchange and the French air traffic control system, were inspired by theoretical ideas and algorithms developed by Fred Schneider, Birman, and others at Cornell. More recently, a long-term collaboration between Birman and Bob Constable led to the automatic verification of the equivalence between optimized and unoptimized protocol stacks. This work built on insights from Susan Owicki and David Gries's work on concurrent program verification.

Our AI group —now our largest group, with nine faculty— is known for its rigorous approach to central problems in AI. Joe Halpern (with Yoram Moses) won the 1997 Gödel Prize for using deep ideas from logic to solve problems in distributed systems. Bart Selman's research, which has led to new methods for solving large-scale reasoning and logistics problems, combines concepts from computation with statistical physics techniques for the study of phase transition phenomena, pioneered by Cornell's Ken Wilson, the 1982 Nobel Laureate in Physics and founder of the Cornell Theory Center.

This booklet illustrates in numerous ways both the traditional science model and its unique expression in computer science.

Our well-known culture of collegiality, nurtured from the start, has made our collective vision for computer science more than the sum of the individual faculty efforts.

Our collegial atmosphere has fostered a continual dialogue among the faculty on teaching, research, the nature of computer science, and the future of the field. In such a fast-moving and diverse field, such discussions have been essential. When faculty from decidedly different research areas talk together over an extended period of time, perspectives change; research broadens; respect increases; at times, joint research is done where it was not previously contemplated; and new subfields emerge at these boundaries.

collegial: Characterized by or having authority or responsibility shared equally by each of a group of colleagues. Characterized by camaraderie among colleagues.

Our culture of collegiality goes back to 1965, with the tradition of the CS faculty lunching together. Discussions over lunch and coffee covered everything from research, teaching, and student admissions to the culture of our new field. Of course, sports, cars, boats, literature, and politics were not ignored. Everyone, from instructor to full professor, voiced their opinions. This forum gave people a chance to meet a colloquium speaker, grill a recruit, or discuss the previous day's speaker.

Frequent faculty discussions, besides leading to mutual respect, have inspired a great deal of interdisciplinary work.

A collaborative, collegial atmosphere led our faculty to interact with faculty from other departments and to embrace interdisciplinary work. We helped create Cornell's Cognitive Studies Program and the Theory Center. We hired computational biologist Ron Elber, who works with Steve Tanksley in Plant Biology and David Shalloway in Biochemistry, among others, on protein structure problems. We helped create the new graduate field of Computational Biology. Keshav Pingali works with Tony Ingraffea in Civil & Environmental Engineering to bring grid-computing ideas to bear on large multi-physics computational science simulations in the aerospace domain. This kind of work is at the core of the new subfield of Computational Science & Engineering, which Cornell is creating.

"Cornell created one of the first (and best) computer science departments … and that early lead has helped the University build exceptional strength in the Information Sciences across several departments."

Report of the Research Futures Task Force, Cornell, '99

Another example of this kind of work is the new $2 million NSF grant for research on petabyte storage devices for database-driven science, led by Alan Demers. It brings together seven CS faculty in databases, graphics, and Web analysis, along with Astronomy's Jim Cordes, who is collaborating on the design and implementation of a data management system for the Cornell Arecibo Telescope in Puerto Rico.

Because of the pervasive use of computing and the need for computer scientists to engage in joint research, we pressed Cornell to create the unique, college-level Faculty of Computing and Information Sciences (CIS), whose sole purpose is to promote computing throughout Cornell. We are now part of CIS, although we are still affiliated with our two traditional homes: the colleges of Engineering and Arts & Sciences. Yes, CS at Cornell was born with interdisciplinary expectations, met them, and now has still broader interdisciplinary expectations.

Our department has always been a leader in education, both on campus and nationally.

We produced texts that influenced the development of the field. For example, John Hopcroft co-authored groundbreaking texts in algorithms and in formal languages and automata theory, David Gries wrote the first text on compiler construction, and Gerry Salton's texts led the field of information retrieval.

In the 1980s, Gries's text brought ideas about formal programming methodology into the undergraduate curriculum. Tom Coleman, Nick Trefethen, and Charlie Van Loan (co-author, Gene Golub of Stanford) wrote influential texts in scientific computing. And the influence continued into the 1990s and 2000s, with texts by Bill Arms, Dexter Kozen, Johannes Gehrke, Jon Kleinberg, and Eva Tardos, to mention a few. On page 44 is the list of books written by our faculty.

A teacher affects eternity; a teacher can never tell where their influence stops.

Henry B. Adams

Teaching has always been central to our mission. Senior and junior faculty continue to create and teach undergrad courses at all levels, and undergrad research is an important component of our curriculum. Research ideas move quickly into the curriculum, students benefit from faculty who are at the forefront of their areas, and our own research derives benefit from the interaction.

Small wonder that Cornell has given exclusive teaching and advising awards to seven of our current faculty members and that CS professor Dan Huttenlocher was recognized as the New York State Professor of the Year (over all disciplines).

Computer science has seen momentous changes since it started in the mid 1960s. One true invariant over these 40 years is the Cornell CS Department's leadership in research and education. We offer this symposium and publication as a celebration of that long and influential success. And we celebrate what distinguishes the department: a passionate dedication to computer science as a coherent discipline with deep synergy between theory and practice, a collegial atmosphere of mutual respect and support, a collaborative, interdisciplinary environment, and attention to education at all levels.

BCS

Dick Conway and Bill Maxwell of Industrial Eng. develop CORC on the Burroughs B-220 and Control Data 1604 to provide a simpler language than Fortran or Algol. CORC can be described on a single page. CORC is taught beginning in Fall 1962.

CS starts with faculty Dick Conway, Pat Fischer, Juris Hartmanis (Chair), Chris Pottle, Gerry Salton, Sid Saltzman, Bob Walker.

1965

The Computer Science Department is formed. (Conway spent his later years in the Cornell School of Management and is retired, and Salton passed away in 1995.) Housed in both Engineering and Arts & Sciences, CS starts with an MS/PhD program.

Gerry Salton brings his SMART system, started in 1961 at Harvard. SMART is his main tool for 35 years of experimental research in information retrieval.

Juris Hartmanis publishes the paper that starts the field of computational complexity, with Dick Stearns: *On the computational complexity of algorithms, Trans. Amer. Math. Soc.* 117 (1965), 285-306. Later, they receive the ACM Turing Award for this work.

CS produces its first PhD, Joel Sturman, a transfer from Electrical Engineering.

Ken Brown, Peter Wegner join.

1966

Juris Hartmanis and Dick Stearns publish the first of many influential texts by CS: *Algebraic Structure: Theory of Sequential Machines* (Prentice Hall).

Roland Sweet, John Hopcroft join.

1967

Dick Conway, Bill Maxwell, and Louis Miller, publish the classic text *Theory of Scheduling* (Addison-Wesley).

Howard Morgan, Alan Shaw, Robert Wagner, Bob Constable join.

1968

Gerry Salton publishes the classic IR text *Automatic Information Organization and Retrieval* (McGraw-Hill).