From the Editor's Desk

David Alan Grier
Editor in Chief

Occasionally, I am invited to a Washington, D.C., meeting on the "Crisis in Computer Innovation." These events follow a pattern. A policy-maker from, say, the Society to Promote Goodness suddenly recognizes a crisis. With his good friend, the Legislative Assistant to Senator Dingbat, Innovation Committee chair, the two plan an "Innovation Summit." They enlist a speaker (such as the Minister of Invention from a Strategically Important Country), persuade a corporation to sponsor an expensive lunch, and invite numerous VIPs: the President of the National Academy of Engineering, the Head of the National Science Foundation, and so on.

Most of the invitees cannot come, so they pass the invitation to equally busy assistants who pass it to a lower-level assistant. Eventually, the invitation reaches equilibrium in each agency, the level at which an employee, believing they can say something important about computer innovation, accepts the invitation. In this case, the luncheon was in an exceedingly pleasant venue with a large veranda and beautifully landscaped garden. I strolled through small clutches of the Washington policy crowd, careful to identify myself as a member of the "Center for International Science and Technology Policy" rather than as a university faculty member. (I have learned that once you introduce yourself as a "professor" all Washington policy-makers lose interest in you.)

As I crossed the veranda, I overheard comments from various deputies and underlings: "Our office has been responsible for all computer innovation since 1946," said one. "We, of course, created the microprocessor," claimed a second. "The Internet," added a third, "was built by our department."

We were soon seated for the luncheon and Senator Dingbat's assistant rose to speak about the innovation crisis when, outside the dining room, a lawnmower was fired up, the noise effectively halting conversation. No one knew what to do. The attendees knew all about innovation and believed that they worked with the cutting edge of technology. Yet, none of them knew how to resolve the situation. It was as if they suddenly exposed the great gap between those who plan for innovation and those who actually do the innovation.

In this issue

Innovation in computer science—as in any field—needs a wide variety of people to support it. Over the years, this magazine has painted a rich picture of innovators in the computer industry. In this issue, we have a broad range of stories about how the industry developed. We start with an article by George Gray and Ronald Smith, regular contributors, about computers originally built by Burroughs that are now Unisys products. Burroughs designed machines quite different from those built by IBM, say, or Univac. We also have a feature about Alan Kay by Susan B. Barnes, which gives us a sense of how an individual researcher contributed to the field.

To explain how universities influenced the computer field, Gopal Gupta's article tells about computer science curriculum developments. Henry Oinas-Kukkonen explores the connections between Vannevar Bush and Douglas Engelbart, and finally, we have the last installment of Mildred Koss's memoirs about working as a programmer in the early computer industry.

In this magazine, we regularly and sadly observe that another colleague has left us. Since our last issue, for example, we have lost John Backus, leader of the original IBM Fortran team and the man who lent his name to the Backus-Naur formalism for describing computer languages. We also must report, especially sad for the *Annals* staff, the passing of the Anecdotes department editor, Laurie Robertson. In addition to her work for the *Annals*, Laurie was an employee of MITRE and contributing, in her own way, to innovation in the computer field.

Closing the gap

At our innovation luncheon, the lawnmower noisily continued. Finally, a staff worker went outdoors and convinced the gardener to delay his mowing, letting us continue our innovation discussion. I was intrigued to see how much time and money each government agency had contributed to computer and software development, and how much they believed that the continued improvement of computers, software, and services was crucial to their mission. It would have been interesting, however, to have had a wider variety of talent at the table: those who built the technology, those who sold it, and those who managed it. I was pretty certain that one of them would have had the sense to go outside and ask the gardener, politely but firmly, to shut off the lawnmower.

Readers may contact David Alan Grier at grier@gwu.edu.