The History of Design in Computing

In this issue of the *Annals*, we are pleased to provide three very different but equally unique contributions to the history of computing.

In their article on “Aspects of the History of Computing in Modern Greece,” Konstantina Dritsa, Dimitris Mitropoulos, and Diomidis Spinellis provide a broad survey of computing in Greece from the 1920s to the present. The first sections of their history focuses on the role of IBM in the computerization of industry and government—among other distinctions, Greece was among the first of the European countries to receive IBM hardware in the 1930s—with a particular focus on the computerization of the National Bank of Greece. In this last case, the role of the American consultants who as part of the post-war Marshall Plan encouraged Greek organizations to adopt American technologies and practices is reminiscent of Corinna Schlombs’s recent *Annals* article on similar dynamics in other countries in Europe. By the mid-1970s, however, Greece was developing its own institutions for computing training and research, and by the early 1980s, the nation was investing in national research and development networks. The overall picture is one of significant but intermittent bursts of computer-related developments in Greece, and in providing such a comprehensive overview of a region often neglected in conventional histories of computing, Dritsa, Mitropoulos, and Spinellis have provided an important contribution to our community.

David Halsted, in his article on the “The Origins of the Architectural Metaphor in Computing: Design and Technology at IBM, 1957–1964,” begins by reminding us of the strangeness of such metaphors. Like most metaphors, it seems only obvious in retrospect, and its meaning and influence can be understood only when it is properly historicized. Halsted focuses particularly on the IBM designer Eliot Noyes, who in 1957 applied concepts drawn from high modernist architecture to IBM industrial design. This was a very specific application of the architectural metaphor, and not at all the simplistic association of two types of construction that contemporary applications of the metaphor often imply. Even when the metaphor was adopted and adapted by fellow-IBMer Frederick Brooks to describe his approach to software development, it still retained its specificity. But as the metaphor spread throughout the computing community, what was meant by “computer architecture” became increasingly divorced from its origins on conventional architecture. In untangling the changing meanings of the metaphor, Halsted mobilizes the history of the words and concepts used to understand the computer to tell us something very significant about the history of computing.

In their article “Less Is More in the Fifties: Encounters between Logical Minimalism and Computer Design during the 1950s,” Liesbeth De Mol, Maarten Bullynck, and Edgar G. Daylight explore the influence of the philosophical tradition of logical minimalism on the design of early computers. Dismissing the simplistic connections often drawn between developments in mathematical logic and computer design via the work of John von Neumann and Alan Turing, these authors focus instead on the contributions of Haskell Curry, a professor of logic at the University of Pennsylvania who, via his involvement in the wartime ENIAC project, developed an approach to reduced-instruction-set programming that was very deliberately an attempt to apply a logical
minimalist approach to computing. This approach made its way to the British National Physical Laboratory and the ACE architecture not so much because of Turing, but because of the influence of Curry’s work at the ENIAC on Harry Huskey, who would later introduce a logical minimalist-inspired design to the ACE computers (and later to the machines he design for Bendix). In the 1950s, interest in logical minimalism as it applied to computing was encouraged by the work of Alonzo Church and Stephen Kleene, and it was through their work that the early contributions of Turing were reinterpreted through the lens of logical minimalism. It was only then that the minimalist approach modeled by the idealized Turing machine was attempted to be implemented in actual real-world computing devices, such as Willem van der Poel’s ZERO machine (1952).

By the 1960s, new innovations in transistor technology made the need to minimize the logical design of computers less pressing from an engineering and manufacturing perspective, but as De Mol, Bullynck, and Day clearly demonstrate, the influence of logical minimalism in the conceptual models used by computer scientists becomes only more significant over time.

ABOUT THE AUTHOR
Nathan Ensmenger is the editor in chief of IEEE Annals of the History of Computing and an associate professor in the School of Informatics and Computing at Indiana University. Contact him at nensmeng@indiana.edu.