From the Editor's Desk

Lars Heide
Editor in Chief

Modern generally denotes something that is up to date, new, or contemporary. Therefore, "modern computing" contrasts with something past, which gets blurred as the technology is developed beyond the accomplishments originally termed "modern." The cover picture and the five articles in this issue of the *Annals* illustrate the use of modern and how its substance has changed since the late 1940s. The cover of this issue depicts a Trans World Airlines (TWA) agent and an early Teleregister ticket terminal in 1961. TWA was a major domestic American airline from 1925, and the main US-based operator on intercontinental routes from 1946, until it was merged into American Airlines in 2001. With this picture, we can see how TWA modernized by introducing online ticketing based on seat reservations stored on a magnetic drum, a predecessor to magnetic disks. The new system replaced ticketing based on paper and telephone calls to the reservation center holding the reservation records. TWA's status as "modern" faded a few years later, however, when in 1964, American Airways introduced IBM's new Sabre system for airline reservation and ticketing that used "dumb" video terminals to communicate with the reservation records.[1] The Teleregister Corporation, established in 1928, produced display units for stock brokers and, in 1952, created a seat management system for the American Airlines' New York office.[2]

In the first article in this issue, "Reconsidering the Stored Program Concept," Thomas Haigh, Mark Priestley, and Crispin Rope discuss why "stored program" and the implementation of the "stored program concept" in the late 1940s was an important dividing line in the history of computing, separating modern computers from their predecessors. The authors document that our understanding of what the stored-program concept is and of why it is important changed significantly as scholars provided new perspectives on what constituted "modern." This article is the first of three articles by the authors accepted for publication in *Annals* that revisit the legacy of ENIAC in the history of computing. The second, "Engineering 'The Miracle of the ENIAC': Implementing the Modern Code Paradigm" examines the conversion of ENIAC to use the "modern code paradigm" defined in this first article. The third, "Los Alamos Bets on ENIAC: Nuclear Monte Carlo Simulations, 1947–1948," examines in detail the first program written in the new paradigm to be executed. The second and third articles in this series will appear in forthcoming issues of the *Annals*.

In the second article in this issue, "Computing in Transition: The Origins of Barcelona's School of Informatics, 1976–1984," Jordi Fornes and Nestor Herran analyze the establishment of the first computer science university degrees in Catalonia in 1976 in the context of Spain's political transition from the Franco dictatorship to a democracy. They show that the development of Barcelona' School of Informatics was the result of policymaking by a powerful association of technicians in informatics that pushed government and academy for formal studies inside university.

The third article is Nicholas Rawlings' "The History of Nomad: A Fourth Generation Language." Rawlings tells the story of the design and implementation of Nomad in the early 1970s. Nomad is a database-oriented programming language that provides both interactive and batch environments for data management and application development, including commands for database definition, data manipulation, and reporting. The intension was to both enable nonprogrammers to write substantial programs themselves and enable programmers to write major applications in much less time than using existing programming languages. Nomad was highly successful, with more than a half a million users in the 1980s, and it is still used today.

Computer scientists and technicians have been developing programming languages since the late 1940s and two decades later, several scholars found reason to distinguish between early and more recent—more modern—languages.[3] Nicholas Rawlings finds Nomad more modern than the programming languages of the late 1960s. Innovation in programming languages have produced successively more "modern" languages. This development has been structured as generations, but several competing numberings exists. (Burton Grad was most helpful in acquiring and editing this article.)

 Published by the IEEE Computer Society 1058-6180/14/$31.00

The fourth article in this issue is "Messaging in the Early SDC Time-Sharing System" by David Hemmendinger. Today, emails are seen as an important element of modern computing. This article tells how an early messaging system grew within the time-sharing computer in the early 1960s. It facilitated communication between individual users, and it also enabled collaborative work such as joint debugging of multiauthor programs, an activity now associated with the Internet. These capabilities appear to be the earliest documented implementation of text messaging and were in use before electronic mail was developed for time-sharing systems.

The last article in this issue is Zbigniew Stachniak's "Early Commercial Electronic Distribution of Software." Stachniak studies the distribution of software for the early personal computers in the late 1970s. At that point, the consumption and distribution of modern computing differed significantly for many small computer users from the software industry's distribution to industry, government, and universities. Stachniak provides an important map of early electronic methods of distribution attempted by several companies in North America and Europe.

References and Notes

1. J. Eklund, "The Reservisor Automated Airline Reservation System: Combining Communications and Computing," *IEEE Annals of the History of Computing,* vol. 16, no. 1, 1994, pp. 62–69; D.G. Copeland, R.O. Mason, J.L. McKenney, "Sabre: The Development of Information-Based Competence and Execution of Information-Based Competition," *IEEE Annals of the History of Computing,* vol. 17, no. 3, 1995, pp. 30–57; R.V. Head, "Getting Sabre off the Ground," *IEEE Annals of the History of Computing,* vol. 24, no. 4, 2002, pp. 3239.
2. "Teleregister. Special Purpose Electronic Engineering That Sets the Pace!" Teleregister Corp.,1956, Computer History Museum, accession 102646324.
3. S. Rosen ed., *Programming Systems and Languages,* McGraw-Hill, 1967; J.E. Sammet *Programming Languages. History and Fundamentals,* Prentice-Hall, 1969.

Lars Heide *is an associate professor at the Centre for Business History at the Copenhagen Business School. Contact him at heide.lpf@cbs.dk.*

IEEE Annals of the History of Computing

Editorial Board