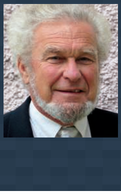
50 & 25 YEARS AGO



EDITOR ERICH NEUHOLD
University of Vienna
erich.neuhold@univie.ac.at

AUGUST 1970

In the early years, *Computer* was only published bimonthly. Therefore we will have to skip our interesting and/or informative extractions for August. The next one will appear in the September 2020 issue of *Computer*, and we hope you will eagerly wait for our next publication of this column.

AUGUST 1995

www.computer.org/csdl/mags/co/1995/08/index.html

Where Is Software Headed?; Ted Lewis et al. (p. 20): "To find out where software is headed, *Computer* took to the Internet, asking experts in academia and industry to share their vision of software's future. Their responses suggest a strong polarization within the software community. … This was supposed to be an introduction to the detailed comments on the following pages. After reading these selections, as well as others that were not chosen, I was struck by the chasm that exists between academia and industry. … The second impression, after realizing that the two groups are on different wavelengths, is the heavy emphasis on programming languages, operating systems, and algorithms by the academic group, in contrast to the clear emphasis on standards and market-leading trends by the industrial group. … The academic group uses words like 'efficiency, difficult problem, and evolution,' while the industrial expert uses words like 'time to market, opportunity, and revolution' to describe their world views. To an industrial person, things are moving fast—they are revolutionary. To an academic, things are moving too slowly, and in the wrong direction—they are only evolutionary changes which are slave to an installed base." *[Editor's note: I am not commenting on the nine contributions that Ted Lewis selected, but I recommend reading them. Their views often correctly predict future developments; but equally often, they are off the mark.]*

SPEC as a Performance Evaluation Measure; Ran Giladi et al. (p. 33): "A computer performance evaluation technique must provide decision-makers added information value. The authors challenge the validity of SPEC measures and offer an alternative approach to obtaining these measures. … Potential computer system users or buyers usually employ a computer performance evaluation technique only if they believe its results provide valuable information. System Performance Evaluation Cooperative (SPEC) measures are perceived to provide such information and are therefore the ones most commonly used." (p. 34) "Since its advent, SPEC has defined five benchmark suites, from which various measures have been derived.[5,6] The SPEC[89] suite included 10 programs, four written in C and the rest in Fortran, representing practical engineering and scientific applications. … In 1992, the SPEC[89] suite was found to be unsatisfactory, so SPEC defined two program suites that each determined a measure, the CINT92 suite for SPECint92 and the CFP92 suite for SPECfp92." *[Editor's note: The article analyzes in detail the performance-measurement principles employed in SPEC program suites and suggests some improvements. SPEC was founded in 1988, and since that time, it has developed numerous such measures for different application environments. SPEC92 has long been retired, but SPEC CPU 2017 looks like a successor.]*

Program Comprehension During Software Maintenance and Evolution; Anneliese von Mayrhauser (p. 44): "For years, researchers have tried to understand how programmers comprehend programs during software maintenance and evolution. Five types of tasks are commonly associated with software maintenance and evolution: adaptive, perfective, and corrective maintenance; reuse; and code leverage. … The program comprehension process uses existing knowledge to acquire new knowledge that ultimately meets the goals of a code cognition task. This process references both existing and newly acquired knowledge to build a mental model of the software that is under consideration. Understanding depends on strategies. While these cognition strategies vary, they all formulate hypotheses and then resolve, revise, or

Digital Object Identifier 10.1109/MC.2020.2993329
Date of current version: 30 July 2020

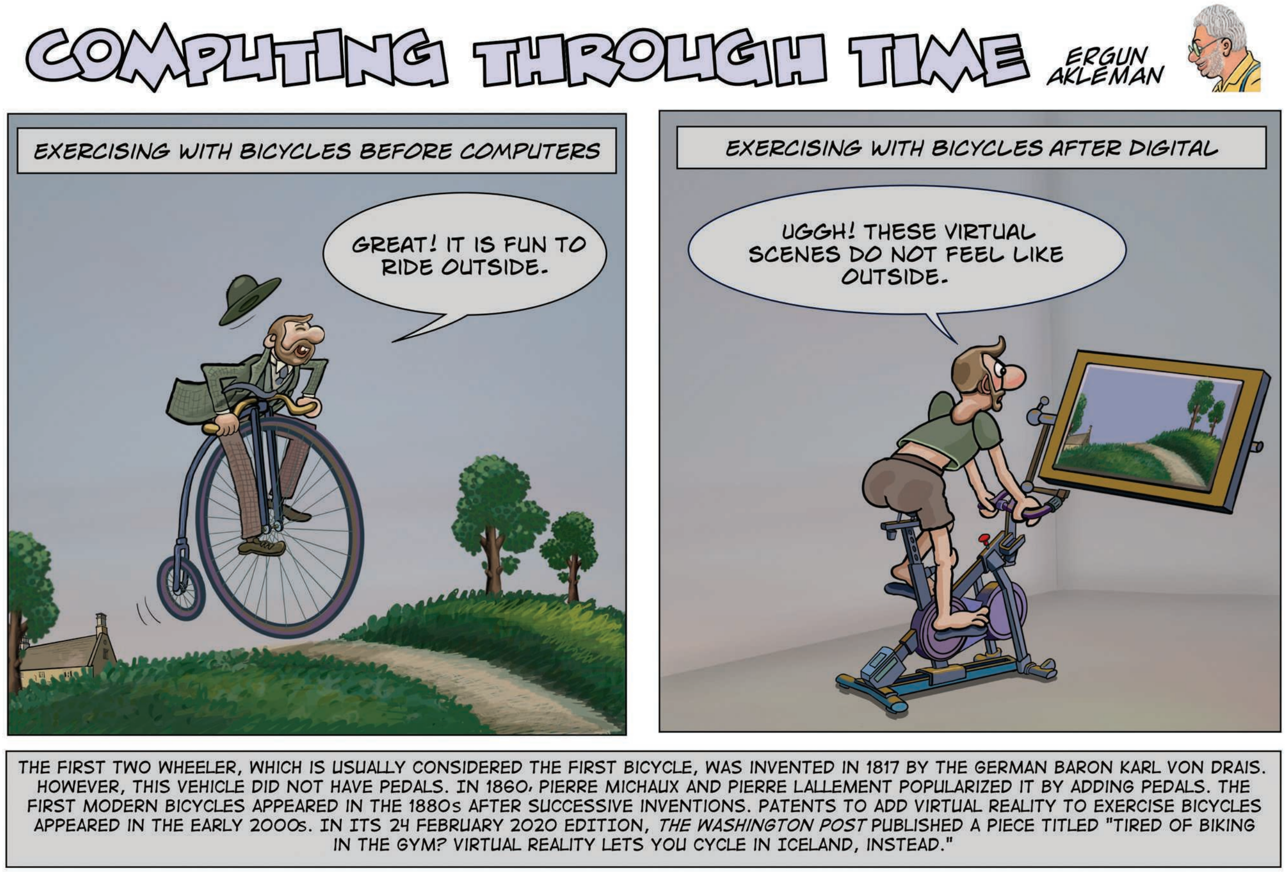
abandon them." (p. 51) "The integrated code comprehension model has four major components: the top-down, situation, and program models and the knowledge base. The first three reflect comprehension processes. The fourth is needed to successfully build the other three." *[Editor's note: The detailed and interesting analysis of six program comprehension models concludes that, clearly, a combination of the various approaches leads to better results. But quite a number of questions still remain as the subject of further research. Actually, I do not believe that the intervening 25 years have provided us with many improvements of the described approaches.]*

Computer Science Research in Mexico; Vladimir Estivill-Castro (p. 56): "Since research is fundamental to a country's human resource development, it is crucial in technological disciplines that evolve rapidly, and no discipline has evolved faster in the past 30 years than computer and information science and engineering (CISE)." (p. 58) "There are two agencies for the promotion and funding of science in Mexico: CONACYT and SNI (Sistema Nacional de Investigadores), both federal agencies. … Neither recognizes CISE research as such, and the criteria for awarding merit could not be less favorable for CISE." (p. 62) "It is possible for less developed countries to create an environment for R&D in CISE. The relatively good health of the economies in Argentina, Brazil, and Chile is fairly recent. Moreover, the shining example provided by the situation of CISE in India indicates that it is possible to orient resources and obtain impressive results." *[Editor's note: I believe that the detailed analysis of the CISE situation in Mexico not only applies to that country but also to other developing countries. Furthermore, it does not only describe the situation in 1995 but also the situation that still exists in 2020. Only India and China (see below) have managed to establish profound research and advanced applications that have worldwide impact.]*

From Nationalism to Pragmatism: IT Policy in China; Kenneth L. Kraemer (p. 64): "IT use in China is still low compared to other countries, but investment in hardware, software, and services has increased about 20 percent annually since 1989 and shows no signs of slowing. Computer hardware production grew 29 percent annually from 1987 to 1993. Given its market potential, its large pool of engineers and computer professionals, and its low-cost labor force, China is likely to become a major producer of low-end computer hardware and Chinese-language software in the next 10 years." (p. 69) "Relative to its population and GDP, China's level of installed computing capacity falls far behind newly industrialized countries and even lags other developing countries in the region. At its present growth rate, it will take decades for China's IT use to catch up with that of its neighbors." (p. 72) "China's policies have shown signs of success, as computer use and production have grown dramatically in recent years. The key to the success of China's technology policy seems to be rooted in pragmatism. Policy makers appear willing to change and adapt when existing policies are not achieving their goals or when new opportunities appear." *[Editor's note: The article identifies many of the successes and shortcomings in IT that existed in China in 1995. However, in its extrapolation, it ignores the huge changes that had already happened during the 1980–1995 period. Here, an explosive industrial base and a highly skilled labor force were created and laid the foundation for the still existing rapid economic growth in China.]*

The Linux Operating System; Shahid H. Bokhari (p. 74): "It was written by Linus Torvalds, a graduate student at the Helsinki University of Technology, Finland. Since he wrote it from scratch, it does not contain proprietary code. He holds the copyright but permits free distribution of the source code (see "The Linux software license" sidebar). In addition to Torvalds, an enthusiastic, worldwide group of volunteers collaborated in developing this operating system through the Internet. … This article attempts to remedy that situation by describing how Linux can be obtained, installed, and used. It is based on my experiences with Linux over the past two years. With no prior experience with Unix system administration, I've installed nearly two dozen Linux systems without serious problems." (p. 79) "Linux is here to stay. It is a stable, powerful operating system that runs on cheap commodity hardware and can be freely distributed. … Most PCs in the world run DOS/MS-Windows and commercial or entertainment applications based on these operating systems. Linux will probably have a minor impact on this market. However, within the scientific, engineering, and academic communities where Unix is ubiquitous, Linux will be a major force to be reckoned with." *[Editor's note: The article correctly predicts the staying power of Microsoft Windows in the PC market but totally underestimates the huge impact Linux has had on many other systems, whether supercomputers, Apache servers, or the Android operating system.]*

What Goes Into an Information Warehouse?; Capers Jones (p. 84): "For the Information Age, we need measures on the cost of creating, using, and transporting information, as well as the cost of finding and fixing errors. We could also use some supplemental facts, such as the ratio of useful to extraneous information, the quantity of information created and destroyed annually, and the relative volume stored in paper form, in magnetic or optical form, or redundantly in multiple forms. … Table 1 summarizes the volume of information stored by our hypothetical firm *(of 250 000 employees)*. Note that the total amounts to more than 1,000 pages per employee, and close to half of it (451 pages per employee) is stored on paper. Given that 250 pages of ordinary 20-pound office paper make a 1-inch stack, the paper information is roughly equal to a stack 37,625 feet high (more than 7 miles)." (p. 85) "Although hypothetical, our case study strongly suggests four topics in need of significant research to develop truly effective data-warehouse concepts and tools: cross-references


COMPUTING THROUGH TIME
ERGUN AKLEMAN
EXERCISING WITH BICYCLES BEFORE COMPUTERS
GREAT! IT IS FUN TO RIDE OUTSIDE.
EXERCISING WITH BICYCLES AFTER DIGITAL
UGGH! THESE VIRTUAL SCENES DO NOT FEEL LIKE OUTSIDE.
THE FIRST TWO WHEELER, WHICH IS USUALLY CONSIDERED THE FIRST BICYCLE, WAS INVENTED IN 1817 BY THE GERMAN BARON KARL VON DRAIS. HOWEVER, THIS VEHICLE DID NOT HAVE PEDALS. IN 1860, PIERRE MICHAUX AND PIERRE LALLEMENT POPULARIZED IT BY ADDING PEDALS. THE FIRST MODERN BICYCLES APPEARED IN THE 1880s AFTER SUCCESSIVE INVENTIONS. PATENTS TO ADD VIRTUAL REALITY TO EXERCISE BICYCLES APPEARED IN THE EARLY 2000s. IN ITS 24 FEBRUARY 2020 EDITION, *THE WASHINGTON POST* PUBLISHED A PIECE TITLED "TIRED OF BIKING IN THE GYM? VIRTUAL REALITY LETS YOU CYCLE IN ICELAND, INSTEAD."

Digital Object Identifier 10.1109/MC.2020.2998616
Date of current version: 30 July 2020

between domains, redundancy between paper and on-line storage, ratio of graphics and images to alphanumeric or textual information, and effective metrics for normalizing information on data volumes and data quality." *[Editor's note: The short analysis finds that, of the information kept by a large corporation, nearly half of it was in printed form. In 1995, that was probably right, but it changed rapidly with the advance of the Internet and the web. There is, however, no analysis concerning what has been kept in paper form and what has been stored electronically, so the question raised in the title remains unanswered.]* C