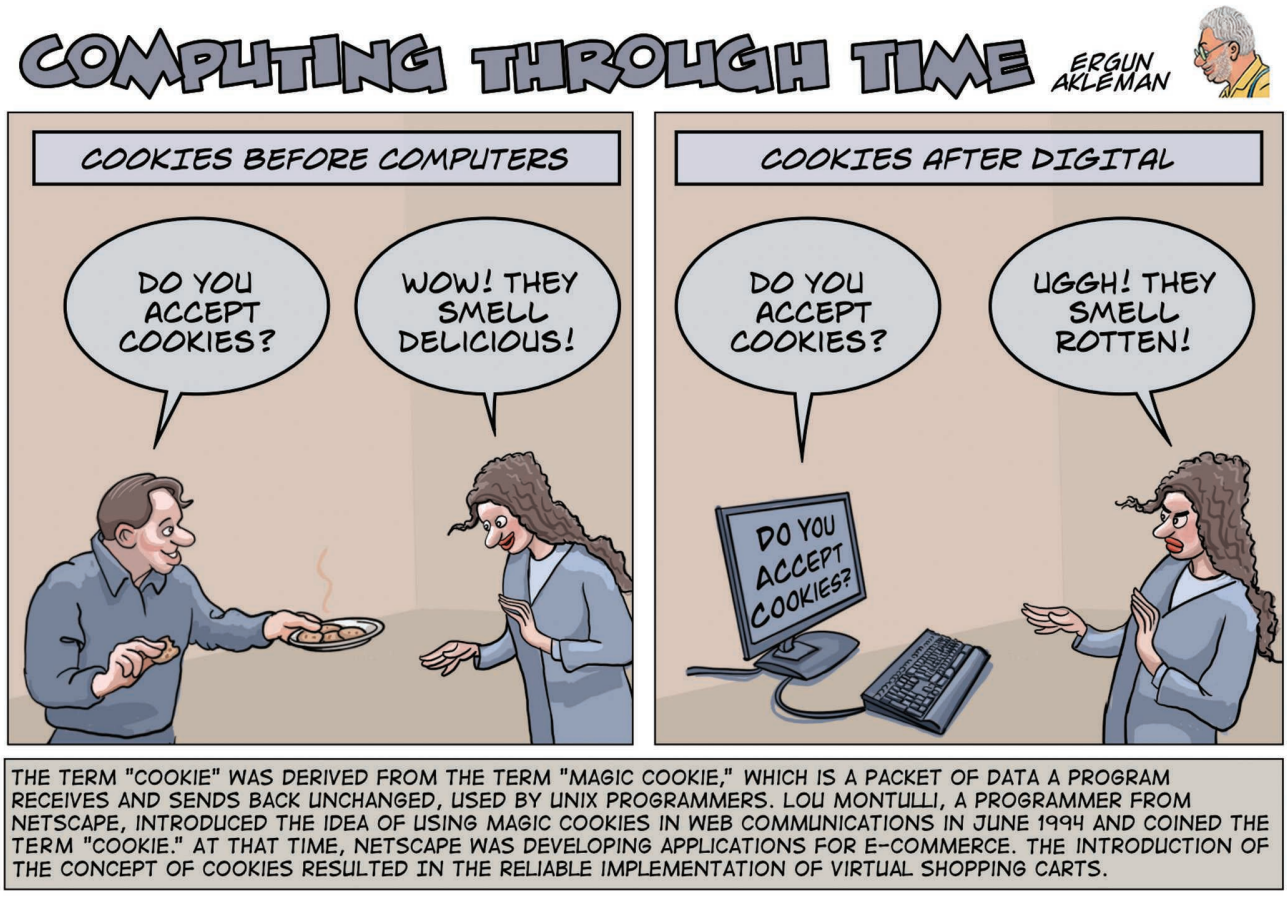

Digital Object Identifier 10.1109/MC.2021.3070567
Date of current version: 4 June 2021

data? Much depends on the system's accuracy—the distance between its results and the desired results—and on the system's precision—the size of the value range it returns." (p. 57) "To satisfy the requirements of both the inexact-agreement problem and the sensor-fusion problem, we merged the optimal region algorithm with FCA to produce an algorithm that provides the best accuracy possible and increases the precision of distributed decision-making." *[Editor's note: This is a very interesting article, especially now in the age of the Internet of Things. It discusses, in depth, a number of already proposed algorithms to solve the problem and then offers a hybrid one that eliminates some of the problems the others have.]*

Creating Printed Music Automatically; Gary M. Rader (p. 61) "The quality of printed music produced by software programs depends on the user's knowledge of copyist rules. This constraint-based technique automates uncomplicated, common notational tasks. ... Unfortunately, we do not have the rules for notating music in a prescriptive form, that is, one that a computer can use. Music copyists take years to learn them." (p. 68) "MusicEase performs justification in two passes. The first calculates the ideal spacing for each note or chord and its elements, plus meter changes, clef changes, and so on, for the entire system. The second calculates how much to shrink or stretch each note's space to create a fully justified system." *[Editor's note: This is an interesting article about the problems with the computerized printing of music sheets. It still requires the user to enter all of the components and properties from the original manuscript manually. The program then interprets these components for the many representation concepts required for a properly printed music sheet.]*

Passion: Optimized I/O for Parallel Applications; Rajeev Thakur et al. (p. 70) "To improve the I/O performance of parallel programs with distributed multidimensional arrays, we have developed a software library called Passion (Parallel, Scalable Software for Input/Output)." (p. 71) "Passion supports two basic models for storing and accessing data—the Local Placement Model and the Global Placement Model—and it provides routines to perform the required I/O on arrays and sections of arrays. In both the Local Placement Model and the Global Placement Model, Passion fetches data from a file, waits for the application program (processor) to perform the necessary computations on that data, and writes the results back to a file if necessary."