OCTOBER 1967
In its early years, Computer was published bimonthly. Please be sure to check back for more 1967 highlights in our upcoming November 2017 issue.

OCTOBER 1992

New Designs for Member Services (p. 4) “To help develop a more integrated set of products and services that are timely and responsive to our members’ needs, the strategic plan adopted last November by the Board of Governors called for the establishment of technical segment committees (TSCs). … A TSC is a ‘matrix’ committee in that its members come from various program boards and technical committees (TCs).” [Editor’s note: Over the years, there have been numerous attempts to increase communication among the many IEEE CS bodies as well as with other IEEE bodies. The attempt outlined in this President’s Message is one example; however, it appears that it was only active for about two years.]

Guest Editors’ Introduction: Object-Oriented Computing (p. 6) “In contrast to the RBP (Rule Based Programming), the object-oriented paradigm (OOP) is a general-purpose programming paradigm that works for a wide variety of applications. Like the RBP, the OOP can be viewed as revolutionary, but it is perhaps more appropriately considered evolutionary. … In OOP, objects are first categorized into classes and organized hierarchically according to their dependency and similarity. Each class comprises a set of attributes reflecting the objects’ generally static properties and a set of routines [in Smalltalk, methods] that manipulate these attributes. Then relations between classes, such as inheritance, are designed.”

Dimensions of Object-Oriented Modeling (p. 12) “Encapsulation and reactivity are essential dimensions of object orientation; they support decentralized abstraction, interaction, and evolution of individual objects. Distribution is a strong (two-way) form of encapsulation, while concurrency reduces computation time and enhances modeling power. These four dimensions—encapsulation, reactivity, distribution, and concurrency—provide a framework for exploring the OOP and a basis for distinguishing it from other programming-language paradigms.” [Editor’s note: This article systematically introduces object-oriented programming’s various features and illustrates what distinguishes it from other programming paradigms.]

Object-Oriented and Conventional Analysis and Design Methodologies (p. 22) “A review was performed that resulted in the selection of six analysis methodologies and five design methodologies. The analysis methodologies were DeMarco structured analysis, Yourdon modern structured analysis, Martin information engineering analysis. Bailin object-oriented requirements specification, Coad and Yourdon object-oriented analysis, and Shlaer and Mellor object-oriented analysis. The design methodologies were Yourdon and Constantine structured design, Martin information engineering design, Wasserman et al. object-oriented structured design, Booch object-oriented design, and Wirfs-Brock et al. responsibility driven design.” [Editor’s note: A number of the techniques described have been quite successful and are still used today, sometimes supported by commercially available software systems.]

Applying “Design by Contract” (p. 40) “Assume you decide to use a routine call for one of the subtasks. This is similar to the situation encountered in everyday life when you decide to contract out for a certain (human) task rather than doing it yourself. … It is not difficult to see how the preceding ideas apply to software construction. If the execution of a certain task relies on a routine call to handle one of its subtasks, it is necessary to specify the relationship between the client (the caller) and the supplier (the called routine) as precisely as possible. The mechanisms for expressing such conditions are called assertions. … The precondition expresses requirements that any call must satisfy if it is to be correct: the post-condition expresses properties that are ensured in return by the execution of the call.” [Editor’s note: The concepts developed...
in this and similar papers have found wide application but haven’t achieved the desired breakthrough yet."

**Architecture of an Open Object-Oriented Database Management System** (p. 74) “We concluded that such applications would be better served by an open, extensible object-oriented database (OODB) management system, whose functionality could be tailored, rather than by a single monolithic database management system. Our conclusions are consonant with other research and development efforts. … This article describes the architecture of the Open OODB system. First, we discuss its requirements, then its computational model, which builds database functionality as an extensible collection of transparent extensions to existing programming languages.” [Editor’s note: The 1980s were the height of OODB systems. The Object Management Group (OMG) was founded in 1989—and still exists today—with the objective of establishing a software architecture that provides interoperating object-oriented tools and services. However, OODBs didn’t conquer the world; instead, object-oriented interfaces were put on top of the much more widely used relational database systems.]

**Object-Oriented Programming Shows Gains among Top Industrial Companies** (p. 93) “Reusability, flexibility, ease of maintenance, and the ability to accurately model businesses were cited as major reasons for the adoption of object-oriented programming by many of the Fortune 100 industrial companies in a recent survey. The applications reported include business, science, engineering, and research; they ranged from two- to three-person pilot projects to companywide mission-critical software projects.”

**OS/2 Version 2.0** (p. 97) “Comparisons between IBM’s new OS/2 Version 2.0 and Microsoft’s new Windows Version 3.1 were inevitable. The two companies had been staunch allies in the not-too-distant past—partners in developing OS/2, once declared the successor to DOS as the PC OS for the 1990s. But several things went wrong along the way, and the two companies have since become rivals in the battle for dominance of the PC desktop environment. Both products shipped in the first part of April 1992, and the press has had a field day comparing them. Too often this has involved marketing hype from both companies, with little regard for the comparison of apples to oranges. If you cut through all this, two basic facts should remain clear: Windows 3.1 is still a 16-bit graphical operating environment ‘shell’ that runs on top of the DOS OS. OS/2 2.0 is a complete, 32-bit OS with a graphical user interface.” [Editor’s note: Of course, we now know who won that competition, despite IBM’s OS having many features that surpassed Windows’s offerings.]

---

**myCS** Read your subscriptions through the myCS publications portal at [http://mycs.computer.org](http://mycs.computer.org)

---

**Looking for the BEST Tech Job for You?**

Come to the Computer Society Jobs Board to meet the best employers in the industry—Apple, Google, Intel, NSA, Cisco, US Army Research, Oracle, Juniper...

Take advantage of the special resources for job seekers—job alerts, career advice, webinars, templates, and resumes viewed by top employers.

[www.computer.org/jobs](http://www.computer.org/jobs)