

Preface

The Application System/400™ (AS/400™) is IBM's new generation of midrange computers. Incorporating many characteristics of the System/36 and System/38, the AS/400 has more power and capability than those predecessors. In this issue, the first six papers provide technical information on the design and structure of the AS/400 and show how IBM Business Partners and the user community performed a vital role in its development. We are indebted to G. R. Falkowski and M. E. Gauthier of IBM Application Business Systems in Rochester, Minnesota, for their contributions to the inception and formation of these six articles.

In the first paper of the group on the AS/400, Schleicher and Taylor present a description of the system from the design viewpoint. They describe the motivation for the system, the requirements and other factors bearing on its design, and the approaches used to implement the design. Among the characteristics of the implementation are a layered architecture, object orientation, distributed processing, and independence of programs, data, and systems. In the last part of the paper, the authors briefly describe some of the system facilities.

During the origination of the AS/400, IBM Business Partners, systems engineers, industry specialists, customers, and IBM marketing personnel were involved in developing and testing this new system and in creating and migrating applications for it. The new operating system, OS/400™, incorporates the system functions of the IBM System/36 and System/38 and implements the principles of Systems Application Architecture (SAA). This approach permits support for the billions of lines of application code that has already been developed by industry and provides AS/400 interconnection with System/36, System/38, the IBM Personal System models, and Systems Network Architecture LU 6.2. Pine discusses two of the innovative processes through which this ambitious plan was carried out: early user involvement and migration invitationals.

Feedback from the early user involvement process into the development of the AS/400 meant that early user experiences could serve to enhance the design and development of the hardware and software during their formative stages. Sulack, Lindner, and Dietz discuss the organizational dynamics of this new operational procedure. An iterative process for design, production, testing, and user verification was integrated with the market analysis and verification programs previously discussed by Pine. Together, these elements coalesced into a process breakthrough which ensured that the aggressive goals set for the AS/400 were achieved.

Certain positive features of the AS/400 architecture and certain beneficial functions of the OS/400 were not introduced in earlier systems because they had been accompanied by poor system performance. Clark and Corrigan discuss design and optimization techniques used in the AS/400 that allow the system to offer excellent price and performance characteristics, along with the benefits of those new capabilities.

One of the key features of the AS/400 is its consistent, easy-to-use interface. The Charland paper delves into the help facility, a very important aspect of the user interface and vital to its acceptability. The essential design concept of the AS/400 help facility is the information module. Help is provided in chunks of information beginning at the current position of the cursor, and further help can be requested by the user in chunks of increasing depth and specificity. The author discusses the extensive testing process used to validate this new help facility. The structure of the help facility is derived from the SAA Common User Access (CUA). As required by CUA, contextual help and a help index are provided. A significant additional feature gives users a word-oriented search capability.

The primary focus of the last paper in the group on the AS/400, by Botterill, is a discussion of the rationale and design approach for the user interface—an interface intended to address the needs of a broad

spectrum of users. The AS/400 system interface is primarily a nonprogrammable terminal interface that is based on advances in object-oriented interface design and is simple to use and understand. It is also consistent across all AS/400 functions and successfully introduces consistency between programmable workstations and nonprogrammable terminals.

In a different but supportive vein, Ten Dyke and Kunz provide an introduction and overview on the subject of object-oriented programming. This style of programming was used for the software structure of the AS/400. In the paper, emphasis is placed on setting the subject in perspective among the many other techniques for programming and within the history of techniques that lead to the current interest in the object-oriented approach. The authors use examples to illustrate systems for, and use of, this approach, including examples from Smalltalk and KEE[®].

Almost everyone using a computer has received messages from the operating system or other programs telling the user about actions to be taken or errors found. When a program is being written, the developer must formulate the method by which the program will present such messages to a user. The paper by d'Arielli describes the use of a message management system in designing and implementing an application program for use on a personal computer. The system simply and effectively manages messages for the application. It reduces the coding work of the developer while providing an improved method for message handling within the program.

In this issue we also recognize the many referees who help us maintain the quality of the *IBM Systems Journal*. They are drawn from academia, from the computer industry, and from within IBM. The list of referees for the last four years is included at the end of this issue. On behalf of the authors, readers, and editors, we thank them for their conscientious efforts.

Gene F. Hoffnagle
Editor