Preface

This issue of the *IBM Systems Journal* reflects our continuing interest in systems and processes for improving the productivity of users and developers. The greatest inhibitors to the effective use of computing resources are the limitations inherent in today's development process, and the problem of usability in the systems that result. It is our hope that the approaches described in this issue will contribute to improvements in both the development and usability of systems.

Beregi reviews the state of the craft of software development from requirements definition through systems design and implementation. For each of the phases of the development process, the author identifies the problems inherent in currently employed practices. He proposes an integrated software development environment that addresses these areas of concern. The author's premise is that the application of design and development techniques analogous to those employed in other engineering disciplines can greatly improve the quality of the product of the software engineering craft.

In establishing service level objectives for users and developers, systems managers are forced to make a number of design tradeoffs. Most often these tradeoffs are between the objective factors of hardware and software cost and the subjective factor of user productivity. Thadhani's paper builds on his previous work through the presentation of the results of a new experiment. This study confirms his prior conclusion that maintaining short response time results in significant productivity improvements; additionally, it draws conclusions regarding changes in work patterns resulting from improved response time, and the effect of these altered work patterns on overall productivity.

The previous work on interactive user productivity has correlated improvements in user transaction rate as a result of reduced response time with user productivity. Although we may embrace this measurement intuitively, Lambert reports on a controlled experiment which confirms that the productivity benefits measured by Thadhani in terms of increased interactions reflect a real improvement and not just a change in work habits. To measure the work product of his study and control groups, Lambert used lines of code produced and a metric called function points that reflects the complexity of the programs written. In each case he found an approximately 60 percent improvement in productivity as system response time decreased from 2.0 seconds to 0.85 second.

Systems designed to support large numbers of interactive users require dynamic storage algorithms to support the allocation and reuse of memory as the demands placed on system resources vary. Bozman, Buco, Daly, and Tetzlaff review the classical approaches to this problem and present the results of their work in designing and implementing an improved storage allocation algorithm for VM/SP.

Studies of the work performed by office principals have shown that direct, interactive communication has advantages over the noninteractive mode in the speed of transmission as well as the amount of information communicated through the intonation of speech. Gould and Boies describe a system, originally developed as a research project, to facilitate such interactive communication.

To ensure that systems meet their usability objectives, attention must be focused on their personal interfaces. Neal and Simons describe a system called

Playback that has been used to evaluate such systems and identify areas for improvement. The Playback system provides for the collection and analysis of data in controlled experiments on system use and documentation. The collection of data is performed without interfering with the subject, and thereby does not jeopardize the result.

Developers of systems have a clear responsibility to ensure that the systems they implement meet usability and performance objectives as well as functional requirements. I hope that the papers presented in this issue will contribute to improved programmer productivity and systems efficiency.

John Lacy
Editor