M COMPUTING

Executive Editor
Richard F. Walters, Ph.D.

Managing Editor
Marsha Ogden
Editorial Assistant
Kristi Hanson
Graphic Artist
Sonya Dewey
Production
Jean Sheppard
Composition
World Composition Services, Inc.
Advertising
Judith Brown

M Computing is published by MUMPS Users' Group trading as M Technology Association, Suite 205, 1738 Elton Road, Silver Spring, Maryland 20903. Phone: 301-431-4070, Fax: 301-431-0017.

FROM THE EDITOR

Window to the World?

by Richard F. Walters


Richard F. Walters

One of our more illustrious German MUG members has often cautioned the M community not to attempt to create an "Eiernlegende Wollmilchsau," which, he tells us, is compounded German for an egg-laying, wool-bearing, milk-giving pig. Those of us from a senior generation might harken back to Li'l Abner's "Schmoo" of the 1950s and 1960s: the animal that could do anything and was just dying (literally) to please.

In the past few years, activity in the MUMPS Development Committee (MDC) has increased, not decreased. It is as though the heady experience of actually being able to *change* a language has inspired heights of creativity that few would have deemed possible in the somewhat conservative members who comprise that august group.

The February MDC meeting required an advance mailing of 675 pages, and reducing the scope of proposals to consider for inclusion in MUMPS is not an immediate goal. Features such as internationalization, transaction processing, and object-oriented extensions may be within the grasp of the evolving language. There are, however, other features that MDC is looking at that take a different approach. I am referring to the extensions that provide interfaces to other standards. Interfacing to other standards is a complicated process. But interfaces that enable M users to gain access to language features available in other standards, would obviate the necessity to incorporate those features into M syntax. Rightfully, then, MDC has wrestled with interface to other standards for many years.

Not all standards are pertinent and some present compatibility problems that will take time for the MDC to work out. For instance, compiled languages present a problem when they come up against some dynamic features of M syntax; conflicts may arise under certain conditions. One early achievement, an interface to the GKS (Graphical Kernel System), may not prove that valuable, since the GKS standard has not been widely adopted in the graphics community. On the other hand, interface to the SQL2 standard is an important milestone that MDC is working on at this time.

As we look to the future, M Technology is going to have to be selective—selective in the features it embraces and selective in the standards that it chooses to target. A feature not yet on the drawing board, but one that needs to be addressed soon, is making the interface a two-way street, so that users of other standards will be able to access M globals and the commands that manipulate those globals.

In this issue, we explore several projects that address different aspects of this interface environment. An exceptionally important field is that of bringing M into the

Windows environments of the various platforms on which M is available. The articles by Kristin Johnson with Steven Lathrop and by Guy Gardner explore different aspects of this interface. Zhang writes in depth on accessing M from other computer languages, but takes us to another level as we look at how the spoken language (Mandarin Chinese or Japanese) influences that interface feature.

As our language evolves, wise M users will stay tuned to the auxiliary "standards" they will be able to tap. They'll make optimal use of the features other standards provide. Finally, the chief benefit of this attention to interface between standards is almost certain to be increased visibility of M Technology to everyone else. We have come a long way since the 1970s. Remember dedicated operating system, single-language environments of early MUMPS implementations? That restrictive environment proved a barrier to accessing M for other users for many years. Interfacing M will show it to be a most able team player.

At last, M should take a rightful place of equality with other powerful national and international standards. There's no need to get all those egg-laying, wool-bearing, milk-giving features into one beast. We will just have to learn how others are doing it, and then share our strengths with theirs. M

Richard F. Walters, Ph.D., is a computer science professor at the University of California at Davis.