Editor's Comments

Academics regularly face the task of identifying worthwhile research topics for themselves and their students. Taste in such matters is an important aspect of a scholar's effectiveness.

What criteria might be used in choosing a research topic in the field of MIS? Aside from one's personal interests — it is awfully difficult to sustain an intense research effort without having a keen interest in the subject matter — there are several criteria that should be considered:

- The research should offer some prospect of having a significant influence on practice or providing important new insights in our understanding of the role of information systems in organizations.
- The work should be sufficiently ahead of current fashion that it affords the researcher an opportunity to stake out an attractive new area of investigation.
- The topic should pose sufficient intellectual challenges to merit a serious research effort.
- The topic should include "researchable" issues that may yield valid, general, and reproducible results.

Naturally, not all academics would agree with this list (or, for that matter, with almost any other assertion). A particularly debatable issue is the delicate tradeoff between relevance and rigor, because important real problems generally defy attempts to impose classical research controls. Special considerations may also constrain the choice of research topic, such as the availability of funding to support certain kinds of research.

One of my own current research interests is the application of new software development techniques, which often come under the general (but ill-defined) rubric of *fourth generation languages.* My experience as an observer of two major development projects — admittedly, a small sample — indicates that the claimed goal of a tenfold increase in development productivity is quite attainable under the right conditions. But the skimpy evidence that we have also suggests that achieving such a huge improvement calls for far more than merely substituting a 4GL for COBOL or one of the other 3GLs; it requires a fundamental change — a *paradigm shift* — in the way we approach the application development process.

Suppose, for the sake of discussion, that an order of magnitude improvement in development productivity is indeed possible. What a revolutionary change that would be! No one can possibly foresee all the effects of such a monumental change, but just for starters think about the following possibilities:

- The vast expenditures made to develop the existing portfolio of 3GL software must be viewed as a sunk cost worth only ten cents on the dollar.
- It will cost less to replace a decrepit 3GL application with a brand new 4GL application than it would to maintain the old program for a year.
- It becomes feasible for a very small team to develop a heavy-duty transactions system — equivalent in complexity to a million lines of COBOL, say.
- A comprehensive integrated system can be built from scratch, rather than attempting the complex trick of integrating new 4GL components with existing 3GL applications.
- Organizations will have the option of exploiting the improved productivity by adding greater functionality and "intelligence" to their information system (instead of using the increased productivity solely to reduce the cost and risk of application development.)

The feasibility of achieving dramatic improvements in software development should hardly be in dispute, since this has already been demonstrated. The relative rarity of such improvement suggests, however, that success requires quite special conditions. The following seem to be common features of the most successful 4GL projects:

1. Development of an entirely new integrated system that encompasses a major portion of the organization's "mainline" activities.
2. Use of a set of powerful 4GL tools that support the full range of functions needed in an efficient, high-volume application.
3. Employment of a small team of extremely capable hands-on developers.
4. Unequivocal, knowledgeable, and decisive management support.
5. Willingness to abandon conventional practice with respect to such project management tasks as documentation, interacting with users, organization of the development team, and setting staff compensation.

At this point we can only conjecture about the nature and extent of the forthcoming changes in development methodology. The issues cry out for rigorous research to provide a solid foundation for progress. Surely, the topic satisfies any reasonable test of good research. It deals with matters of fundamental importance to the field of MIS that could lead to substantial rethinking about how we develop management information systems; it could fill in some critical gaps in our knowledge of how we can best employ the emerging development methodologies; and it includes enormously complex technical, economic, and behavioral problems that should challenge the very best minds in the MIS field.

I have little personal doubt about the revolutionary changes ahead in application development and the profound effects that these changes will have on MIS practice. But I could, of course, be quite wrong. Solid research would go a long way toward resolving the matter.

James C. Emery