Application generators: an introduction

by JERROLD M. GROCHOW
American Management Systems
Arlington, Virginia

ABSTRACT

Application generators represent a new class of software development tools, which may yield the next order-of-magnitude productivity improvement in systems design, programming, and maintenance. This paper provides an introduction and bibliography for the topic. Current references to more than 50 articles and publications, as well as to some two dozen products, indicate the extent of recent interest in this topic.
INTRODUCTION

As recently as 18 months ago, an article in a major computer publication stated that application generation was a technology that was unlikely to have any significant impact on the development of computer systems. Today, almost 50 companies have products which they call application generators (see product description bibliography). They operate on a wide variety of computers ranging from microprocessors to large-scale mainframes. Almost every major manufacturer of computers has or is working on application generator products. IBM, with perhaps the largest research and development budget of all, is devoting substantial resources to expanding its offerings in this area. In short, we have seen a revolution within the past two years in the types of software-development tools that are being offered to the industry. Rather than producing just new computer languages, software suppliers are aiming their sights much higher at a new generation of systems software that transcends the concept of both procedural and nonprocedural languages and takes us into the new realm of complete application specification and application generators.

Application generators are a natural outgrowth of our search for better ways to develop not just single programs but complete application systems. While the term “application generator” is currently applied to a wide range of products, it is generally used to connote a development tool or tools whose input is a specification not just of a single program but of an entire system, including the database, transaction formats, reports, programs, and job-control logic. In the ideal case, the output of an application generator would be a complete application system in executable form. However, the current reality is somewhat less than the ideal. Application generators today typically produce only pieces of the application system, often ignoring one or more of the inputs listed above and the consequent outputs.

Claims have been made that application generators result in 10, 100, or even 1,000 to 1 productivity improvements over traditional system-development techniques. While these claims are largely unsubstantiated and are certainly not directly comparable, they are indicative of the goals of the developers of these system-development tools and of the order-of-magnitude changes in the system-development process that can be achieved, at least in certain well-defined cases. Application generators do indeed represent a generational step in the software evolution process.

The following bibliography indicates the range of recent publications on application generators and related topics. A separate section provides a representative listing of products advertised as members of the application generator family.

BIBLIOGRAPHY


39. Shoor, Rita. “Mark V’ Hikes Productivity 50%.”


**REPRESENTATIVE PRODUCT DESCRIPTIONS**


5. Cincom Systems, Inc. *MANTIS.* Cincinnati, Oh.


15. IBM. *Development Management System IVS.* GH20-1863. White Plains, N.Y.


