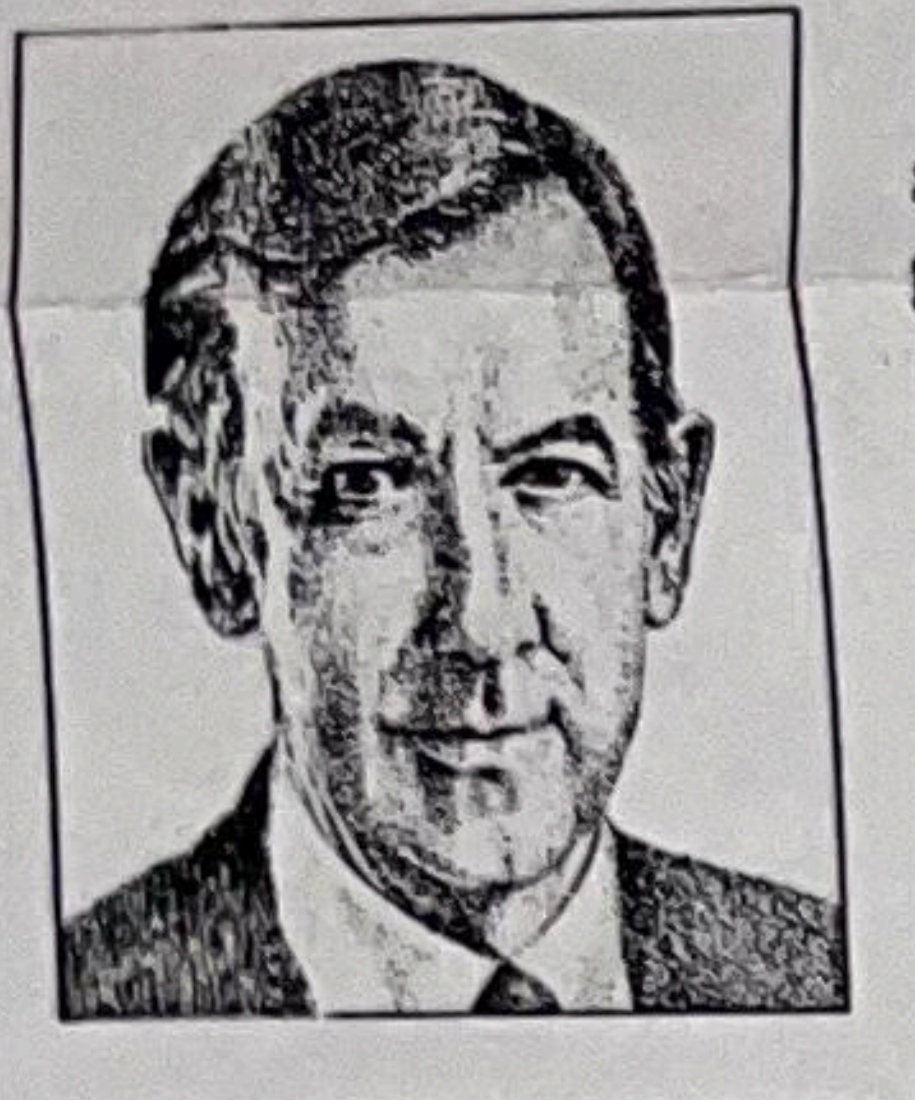
APPLIED INTELLIGENCE

CASE Tools To Play a Larger Role in IS Organizations



JAMES MARTIN

As information systems (IS) organizations retool throughout the 1990s, computer-aided software engineering (CASE) and integrated CASE tools will play a significant role in the process.

Rather than coding applications by hand, IS organizations will become increasingly dependent on integrated CASE (I-CASE) tools to generate code automatically from high-level design specifications.

There are only a handful of I-CASE tools available today. Since the technology is still in its early stages, these tools are incomplete—the offerings lack traceability features, as well as the ability to support C code, create real-time applications and generate applications that can run in Unix target environments.

CASE tools have their roots in the structured techniques that debuted in the mid-1970s. Users manually created and maintained paper stacks of data-flow and decomposition diagrams. After a while, it became exceedingly difficult to keep the diagrams current and to verify consistency across diagram types.

Enter the early generation of CASE tools, which primarily automated the production and maintenance of these structured diagrams. The "CASE pioneer" tools permitted data-flow, decomposition and entity-relationship diagrams to be sketched on the face of the PC. Later, they were expanded to incorporate automated techniques that verified the consistency and completeness of the diagrams.

Now there are more than 200 CASE tools on the market, but most do not fully satisfy the demands of a complete application-generation cycle.

As shown in the diagram, non-integrated CASE tools consist of planning, analysis and design workbenches, a design analyzer and a local repository. The workbenches are used to enter design specifications, in graphical form, into the local repository. The design analyzer checks the specifications for logical consistency and completeness.

Several leading non-integrated tools—such as Excelerator from Index Technology, CASE*Method from Oracle Corp. and Teamwork from Cadre Technologies—generate code for fragments of an application. In most cases, the code generated is limited to screen, report and database-schema definitions. The remaining portions of the application—including the specification of customized procedural logic—must be coded by hand using another product, such as COBOL.

When portions of an application are coded outside of the CASE tool, two repositories of design information are created: a local repository that typically stores non-procedural design specifications derived from the planning, analysis and design workbenches, and a repository that houses the procedural logic coded outside of the CASE tool.

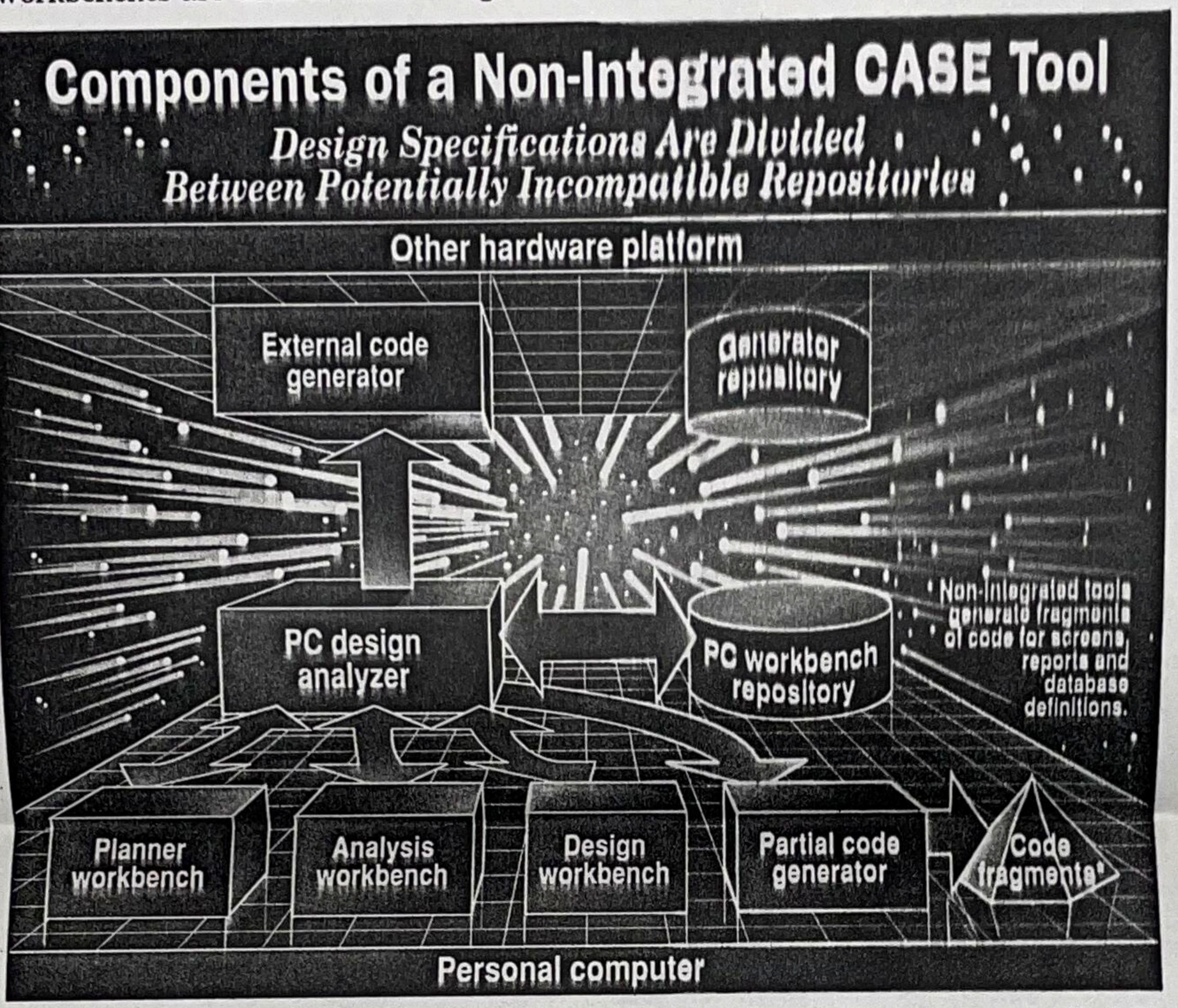
Generally, there is no automated technique to ensure the logical consistency of these two design repositories. The task falls on the shoulders of program managers, who rely on manual techniques to get the job done.

To fill in this hole, many non-integrated CASE tools incorporate a bridge to an external code generator, as shown in the figure. For example, Excelerator has a bridge to Telon, which is used to specify and generate procedural logic; Teamwork has a bridge to Ingres, which is used to generate database schemas.

The primary difficulty of this approach is that, once again, it results in the creation of two repositories—design information contained in the repository of the CASE tool, and the procedural code or database schemas generated by the external tool.

The new architecture of integrated CASE tools solves this problem. These products use a single repository that is tightly integrated with both the front-end design facilities and a back-end code generator. I-CASE tools thus are capable of generating entire applications from design specifications. The result is that they provide the greatest return to the organization for an investment in CASE technology.

Next week, I will discuss the components of I-CASE tools in more detail and the significance of these products in retooling IS. ■


Components of a Non-Integrated CASE Tool
Design Specifications Are Divided Between Potentially Incompatible Repositories
Other hardware platform
External code generator
Generator repository
PC design analyzer
PC workbench repository
Non-integrated tools generate fragments of code for screens, reports and database definitions.
Planner workbench
Analysis workbench
Design workbench
Partial code generator
Code fragments
Personal computer

John Avakian

The concepts embodied in this article are described in the High-Productivity Technology volume in The James Martin Report Series. For more information on this volume, call (617) 639-1958. For information on seminars, contact (in the United States and Canada) Technology Transfer Institute, 741 10th St., Santa Monica, Calif. 90402 (213) 394-8305. In Europe, contact Savant, 2 New St., Carnforth, Lancs., LA5 9BX United Kingdom (0524) 734 505.

The COBOL Programmer Workstation and Its Impact on Productivity

The Micro Focus Developers Seminar Schedule

The Programmer Workstation environment uses personal computers as intelligent, distributed workstations for developing, testing and maintaining host-based COBOL applications. At these Developers Seminars you will:

- See the positive impact the programmer workstation can have on programmer productivity
- See an in depth technical demonstration of the Micro Focus COBOL/2 Workbench facilities
- Watch CICS and IMS code offloaded from the host, running under the integrated Workbench testing environment
- Evaluate the workstation's potential in your organization
- Learn about latest product developments and future trends

July 10th	**Indianapolis, IN**
July 12th	**Dallas, TX**
July 12th	**Buffalo, NY**
July 19th	**Nashville, TN**
July 24th	**Arlington, VA**
Aug 7th	**Cincinnati, OH**
Aug 9th	**Boston, MA**
Aug 14th	**Los Angeles, CA**
Aug 14th	**Charlotte, NC**
Aug 21st	**Chicago, IL**
Aug 23rd	**Pittsburgh, PA**
Aug 28th	**Salt Lake City, UT**

MICRO FOCUS
A Better Way of Programming™

There is ***no charge*** for attending a Micro Focus Developers Seminar. For more information about the Developers Seminar or about Micro Focus products call **415-856-4161**.