

A Dynamic New  
Formula For High  
Quality Compilers:

**The Philon Approach™ = Fast Execution + Portability**



**PHILON™**

## The Solution

*“A product that uses state-of-the-art techniques for building systems software.”*

### *A Product Designed With The Programmer And End-User In Mind: Fast To Develop And Fast To Execute*

The Philon Approach is a highly effective and cost-efficient system which uses state-of-the-art techniques for building systems software. It is comprised of:

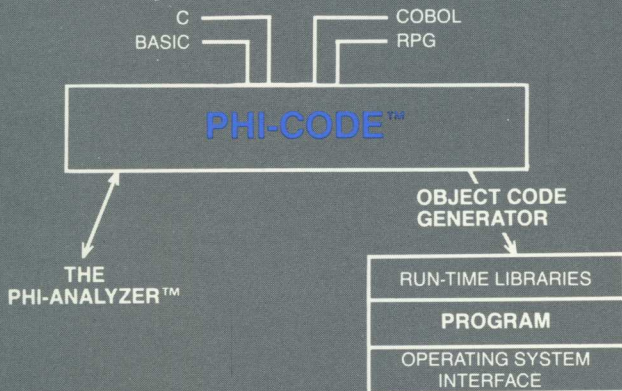
- *A Completely Integrated System* of compiler front ends, debuggers, code generators, operating system interfaces, and run-time libraries, housed in . . .
- *A Totally Portable Environment* capable of being moved from one hardware/operating system to another, boasting . . .
- *A Highly Efficient System* that affords the end-user unprecedented speed of execution plus development features generally offered only by mainframe compilers, including:

- Extensive semantic checking
- Easy-to-understand error messages
- Quick location and correction of programming errors
- Testing tools
- Inter-language subroutine linkage
- A high-powered debugger

The Philon Approach is produced and maintained by a staff of highly-trained professionals. They are dedicated to the ongoing support of their products and clients.

## The Philon Approach

### THE PHILON APPROACH



A Unique Compiler Capability To Meet The Needs Of Hardware Manufacturers, System Integrators, Application Suppliers, And The Government

### How It Works: Phi-Code

The Philon Approach provides an integrated method of developing language systems and uses common intermediate code called Phi-Code.

This Phi-Code is environment independent and may be used to generate the object code for the target machine. Phi-Code can also be interpreted by a high-level debugger—the Phi-Analyzer—which references and displays the program source code.

The Object Code Generator accepts the Phi-Code and produces optimized object code. The final program is surrounded by its own environment, including the operating system interface and run-time libraries. Only the object code generator and the operating system interface need be modified for each hardware/operating system environment that is chosen—a relatively simple task.

As a result, the full complement of Philon compilers can easily and cost-effectively be moved to a new environment.

### How It Debugs: The Phi-Analyzer

The Philon Approach includes a high-powered debugger of unexcelled efficiency: the Phi-Analyzer.

Serving as a common debugger across all Philon languages, the Phi-Analyzer provides these functions:

- **Single-Stepping**  
Executes Phi-Code one statement at a time
- **Variable Monitoring**  
Continuously displays the values of programmer selected variables

- **Profiling**  
Counts the number of times a statement is executed
- **Breakpoints**  
Stops execution on a particular statement, when a given variable is modified, or when a particular programmable condition is met
- **Variable Modification**  
Permits variable values to be modified at breakpoints

The Phi-Analyzer operates on the intermediate Phi-Code and displays each source statement as the program executes. Additionally, the Phi-Analyzer can be used in a "quick compile" mode without object code being generated. This saves the programmer considerable time in testing and debugging.

## The Challenge

*“A product that brings the capabilities of mainframe compilers to the micro and mini environments.”*

### How To Provide Programmers With Fast Executing, Efficient—and Portable—Compilers

For years, the challenge to the industry has been clear: design an integrated series of very fast executing compilers that will be portable over a wide range of hardware/operating system environments . . . a product that brings the capabilities of mainframe quality compilers to micro's and mini's. And develop an approach that must provide:

- Fast execution of programs
- Full instruction sets and additional capabilities that assist the programmer in developing efficient application systems.
- Extensive and easy-to-read error messages, in addition to an interactive debugger, help the programmer save time.
- Compilers that can be easily modified and maintained

In 1980, Philon, Inc. was formed to meet this specific challenge. The result: a singularly effective solution—The Philon Approach.

## The **PHILON** Approach . . .

An essential system to serve both your current and future needs,

Supported by a highly trained, dedicated team of professionals.

☑ Designed for Fast Program Execution . . .

☑ Designed for Portability . . .

☑ Designed for the Programmer . . .

☑ Designed for You!

**PHILON**

---

Philon, Inc.  
641 Avenue of the Americas  
New York, NY 10011  
(212) 807-0303

For more information about how you can profit from The Philon Approach, complete and return the attached convenient reply card. Or call:

**Bob Gildenberg**  
Vice President of Marketing

**PHILON, INC.**  
(212) 807-0303

**YES!** I am interested in learning more about how The Philon Approach can maximize my current and future application development productivity.

- PHILON FAST/BASIC-C
- PHILON FAST/BASIC-M
- PHILON FAST/COBOL
- PHILON FAST/C
- PHILON FAST/RPG

Name \_\_\_\_\_  
Title \_\_\_\_\_  
Company \_\_\_\_\_  
Address \_\_\_\_\_  
City \_\_\_\_\_ State \_\_\_\_\_ Zip \_\_\_\_\_  
Phone (     ) \_\_\_\_\_  
Computer(s) \_\_\_\_\_  
Operating System(s) \_\_\_\_\_