Multi-Language Family

Featuring:

- Family of Integrated Languages and Productivity Tools
- Compatibility Through Industry Standards
- Popular Extensions
- Powerful Debugger and Editor
- High-Performance Features
- CareWare Support Services
- Clear, Accurate Documentation

LPI develops and markets an entire family of high-performance software development tools for the M680X0 series, Intel's 80386 and 80486, AT&T's WE32XXX, the Motorola 88000 and Sun's SPARC RISC-based processors running UNIX®, XENIX®, and DOS.

A Comprehensive Family

The LPI family of compilers includes all of the most widely used languages: BASIC, C, COBOL, FORTRAN, Pascal, PL/I, and RPG II. All are true compilers that provide fast execution speeds.

LPI compilers are designed using LPI's unique Component Architecture technology, which allows cross-language calling; works with a common run-time library, a source-level debugger, and a language-sensitive editor; and permits rapid availability of languages on new processors.

Standards

All languages adhere to or exceed applicable ANSI and X/Open standards and have been validated by the U.S. government. This compliance ensures program portability and protection for your software investment.

Language	Standard
BASIC	ANSI X3.60-1978
True BASIC 87	ANSI X3.113 1987
NEW C	ANSI X3J.11 X/Open
COBOL	ANSI X3.23-1985, 1974 X/Open
FORTRAN	ANSI X3.9-1978 MIL-STD-1753 X/Open
Pascal	ANSI/IEEE 770 X3.97-1983 and ISO LEVEL 0
PL/I	ANSI X3.74-1980 General Purpose Subset
RPG II	IBM S/34/36

Extensions

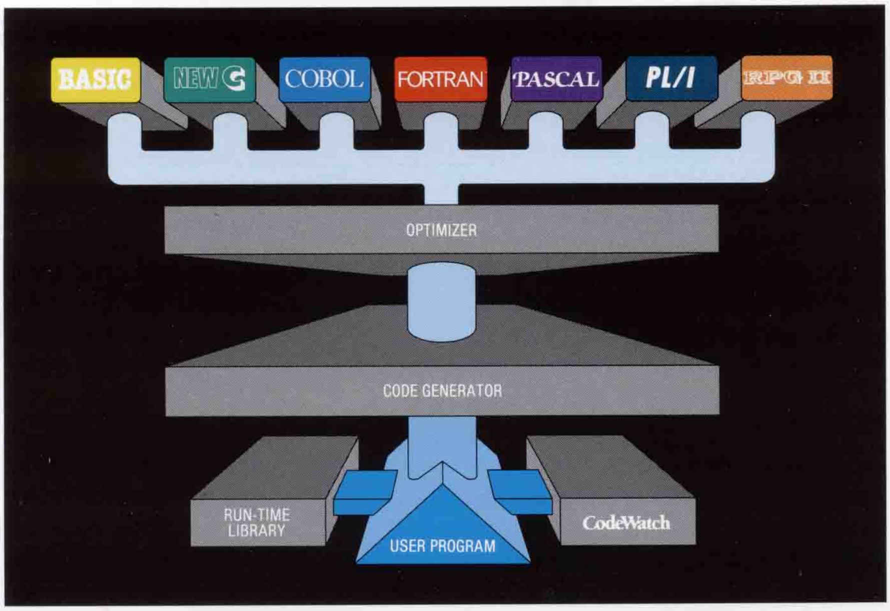
LPI languages also provide a range of extensions for increased functionality and compatibility. These allow you to easily migrate applications written in other dialects to new environments.

Language	Extension
BASIC	CBASIC MBASIC
NEW C	K&R PCC
COBOL	RM/COBOL MF LEVEL II
FORTRAN	VAX-FORTRAN FORTRAN-66 UNIX access
Pascal	Industry Standard Extensions
PL/I	IBM-PL/I VAX-PL/I
RPG II	OCL, GSORT, DFU, WORKSTN, SEU, MGBLD, SFGR

Productivity Features

LPI products contain numerous features that enable the developer to produce quality applications as fast as possible.

Complete error messages fully support the developer's needs. The



This Component Architecture diagram illustrates the flexibility of LPI's modular approach to creating compilers.

different classes of error message, based on severity, are designed to provide the most specific information for the error. They identify the error condition, severity, location, and probable solution. Easy isolation and correction of errors ultimately results in faster completion of development projects.

Component Architecture is the technology underlying each LPI compiler. Four subsystems are combined into each language compiler: the front-end, optimizer, code generator, and run-time library. The advantage of this architecture is that components of each compiler are shared, thereby maximizing portability and code reliability.

Another advantage is that each language front-end is identical across all machines, so developers can easily move their applications from machine to machine, simply by recompiling.

Component Architecture also enables the developer to debug programs written in different LPI languages with a common source-level debugger, CodeWatch.

LPI's unique Code Generator Generator technology allows a high-performance, reliable family of compilers to be provided for a new processor in a shorter period of time and at a lower cost than with traditional compiler technology.

From a description of the characteristics of a new processor, Code Generator Generator technology allows the semi-automatic production of a new code generator. The entire language family can then be easily ported to the new system.

Cross-language calling improves application effectiveness by allowing faster completion of development projects. Programmers may integrate new LPI languages into a program while maintaining existing subroutines. Developers may also target the best language to the given programming task. For example, the numerical processing tasks of an application can be written in LPI's FORTRAN and can be accessed by a main program written in LPI's PL/I .

Cross-language calling can also be extended to native C programs, since LPI compilers produce object code readable by the system linker. This means developers may easily integrate other applications such as DBMS or graphics packages into applications developed with LPI products.

Performance

All LPI languages are true compilers, producing highly optimized object code, that is both fast and compact. This allows the most efficient use of the host hardware.

The LPI compilers produce machine language, not interpretive code. Machine code executes faster. And faster code means better performance for your applications.

Optimizations are both global across entire program units and local within individual statements and expressions. This is provided through the language-specific front-end, a common optimizer, and peephole optimizations in the back-end.

Optimization levels range from no optimization, for fast compilations and debugging, to full optimization, which generates very efficient run-time code.

Specific LPI optimizer features include constant folding, loop induction, dead code elimination, and common subexpression elimination.

Productivity Tools

LPI's productivity tools, CodeWatch and CoEdit, work with the language products to provide an integrated programming environment. LPI languages also work with industry standard editors such as **vi** and **edt**.

CodeWatch

CodeWatch is a powerful source-level debugger that operates with LPI's BASIC, NEW C, COBOL, FORTRAN, Pascal, and PL/I.

CodeWatch dramatically accelerates development efforts by allowing testing and debugging in the language of the source program.

During a debugging session, programmers can determine logic errors and test the required changes. CodeWatch features include conditional breakpoints with action lists, tracing, watchpoints, macros, stepping, and examination and modification of variables.

CodeWatch can debug programs written in multiple LPI languages. It recognizes the language of each module and interacts in the conventions of that language. Only one debugger is required for all LPI languages on each system.

CoEdit

CoEdit is a powerful language-sensitive editor that works with both LPI languages and other products. It features comprehensive editing and pop-up menus, unlimited windows and buffers, keyword templating, context-sensitive online help, and automatic background saving.

Languages:

BASIC

This product is a true compiler and a full implementation of ANSI Minimal BASIC X3.60-1978. Popular extensions from Microsoft's BASIC™ and Digital Research Inc.'s CBASIC® have been incorporated. Programs compiled with LPI's BASIC are fast and compact. Thus, large applications written in BASIC execute much faster than with traditional interpreters.

LPI True BASIC 87

LPI True BASIC-87 is also available. This product is an implementation of the ANSI BASIC X3.113-1987 Standard and offers modern, structured programming capabilities. The MS-DOS® version offers additional features including a full-screen editor, graphics capabilities, a source-code debugger, additional libraries, and a run-time package.

NEW C

NEW C is a fully conforming implementation of the 1988 ANSI Standard for C and is X/Open compliant. It contains numerous extensions that enable developers to use their existing Kernighan & Ritchie and PCC programs. In addition, standard libraries and header files, function prototypes, and new preprocessing directives are included. NEW C contains all the features of the LPI Multi-Language Family: CodeWatch and CoEdit support, complete error messages (over 200 of them), user-selectable optimizations, and cross-language calling.

COBOL

LPI's COBOL is a full implementation of COBOL-85 and 74, validated at the High level, with no discrepancies. It is X/Open compliant.

COBOL extensions that facilitate the conversion of applications written using Micro Focus LEVEL II COBOL™, RM/COBOL, IBM/370®, and the earlier COBOL-68 standard have been implemented.

COBOL's indexed file support is handled by the built-in C-ISAM® indexed sequential file handler. This means that sharing files between your COBOL application and other products that also use the C-ISAM format is possible.

Interfaces to ORACLE®, UNIFY®, and Informix® relational database management systems are managed through C routines.

> ***" You're going to get spoiled quickly.... CodeWatch is one of the finest debuggers I've used. "***
>
> ***-Ross Greenberg***
> ***Editor, UNIX Today!***
> ***July 24, 1989***

FORTRAN

LPI's FORTRAN is a full implementation of the ANSI X3.9-1978 Standard (FORTRAN-77), is GSA validated with no discrepancies, and is MIL-STD-1753 and X/Open compliant. It includes key extensions for compatibility with VAX™ FORTRAN and FORTRAN-66 applications. These extensions ease the transporting of applications from large mainframes to super-microprocessors. Additional extensions allow the developer to take full advantage of the flexibility and portability of the UNIX environment.

As with other LPI languages, an array of user-selectable optimizations is available.

Pascal

This Pascal fully implements ANSI and ISO Level 0 standards. This true, high-performance compiler also includes popular extensions from other dialects of Pascal. LPI's Pascal supports external subprograms and allows developers to compile Pascal program units separately. Developers easily create and maintain modular programs.

PL/I

LPI's PL/I conforms to the ANSI PL/I Subset G standard, assuring you of software investment protection. It includes key extensions for compatibility with super-microprocessor dialects of PL/I, including IBM and VAX versions. Thus, applications running on these systems can be ported to LPI's flexible UNIX-based environment.

RPG II

This product emulates the IBM System/34/36® RPG II environment. It allows developers and end-users to port and run their existing IBM applications on UNIX-based systems.

September, 1989

LPI's RPG II provides a high-performance compiler, OCL processor, WORKSTN file support, and numerous utilities that aid in transporting RPG II programs from IBM systems to multi-user, multi-tasking UNIX-based systems.

Utilities include a Screen Format Generator, Message File Builder, Data File Utility, Sort Utility, Source Edit Utility, Terminal Definition File Utility, and an EBCDIC-to-ASCII File Conversion Utility.

Dedicated Support

All LPI products are provided with thirty days of complimentary introductory CareWare Service. After this period, three levels of increasing CareWare Services are available for your technical support needs. Services range from technical bulletins and assistance with installation, to consultation for large-scale conversion efforts.

Technical Documentation

LPI sees technical documentation as a crucial element of a quality product. The documentation is clear, concise, and accurate. Most documentation sets consist of a language reference manual, a user's guide, a quick reference guide, release notes, and periodic technical bulletins.

The materials contained herein are summary in nature, subject to change, and intended for general information only. Details and specifications regarding specific LPI software are available in the applicable technical documentation.

All rights reserved. LPI, the logo, Language Processors, CoEdit, Component Architecture, Code Generator Generator Technology, LPI-BASIC, LPI-C, LPI-COBOL, CodeWatch, LPI-PL/I, LPI-FORTRAN, LPI-PASCAL, and LPI-RPG II are trademarks of Language Processors, Inc. CareWare is a servicemark of Language Processors, Inc. Microsoft, MS-DOS, MBASIC, and XENIX are registered trademarks of Microsoft Corporation. LEVEL II COBOL is a trademark of Micro Focus, Ltd. RM/COBOL is a registered trademark of Ryan McFarland Corporation. CBASIC is a registered trademark of Digital Research, Inc. IBM, System/34/36 and IBM/370 are registered trademarks of International Business Machines Corporation. SPARC is a registered trademark of Sun Microsystems, Inc. C-ISAM and Informix are registered trademarks of Informix Software, Inc. ORACLE is a registered trademark of Oracle Corporation. True BASIC is a trademark of True BASIC, Inc. Unify is a registered trademark of Unify Corporation. Intel is a registered trademark of Intel Corporation. UNIX is a registered trademark of AT&T. VAX is a trademark of Digital Equipment Corporation. X/Open is a licensed trademark of X/Open Group Members. Copyright © 1989, Language Processors, Inc. Printed in U.S.A.

Language Processors, Inc.

Corporate Headquarters
959 Concord Street
Framingham, MA 01701-4613
Telephone: (508) 626-0006
Telefax: (508) 626-2221
Telex: 951-671

Federal Government Office:
Calverton, Maryland
Telephone: (301) 595-2558

Nippon LPI, Tokyo, Japan
Telephone: 03-297-3561

Ryan McFarland Corporation

Corporate Headquarters
8911 Capital of Texas Highway N.
Austin, Texas 78759
Telephone: (512) 343-1010
Telefax: (512) 343-9487

Sales Office
San Jose, California
Telephone: (408) 241-4800

Ryan McFarland Ltd.
London, England
Telephone: 44(0)-1-799-2434