



PATHSCALE™ EKO COMPILER SUITE: C, C++ & FORTRAN 9x COMPILERS FOR THE AMD64 PROCESSOR FAMILY

PERFORMANCE AND COMPATIBILITY

The PathScale™ EKO Compiler Suite represents the highest-performance 64-bit compiler for the AMD64 processor family, such as the AMD Opteron. This advanced compiler suite takes advantage of the AMD64 high performance 64-bit features and its remarkable price/performance leadership.

EKO stands for "Every Known Optimization" and refers to a compiler framework purpose-built for inserting new optimization techniques to improve performance. At PathScale, poor compiler optimization is considered a bug and a challenge to our development team.

The PathScale EKO Compiler Suite version 1.0 includes:

- C, C++, and Fortran Compilers
- Industry leading optimizations
- Complete support for 64-bit compilation
- Beta support for 32-bit compilation
- Code generation for AMD64 ABI and AMD Opteron
- Compatible with GNU/gcc tool chain and debuggers

The PathScale compilers are derived from the proven and mature SGI®/Cray® compiler suite. PathScale is the first and only company to take this compiler, modernize it to current standards, and enhance the code generator to support the AMD Opteron processor family. The result is the world's highest-performance compiler and complete compatibility with the GNU tool chain.

In real-world application testing, HPC end-users have experienced up to 40% performance gain over alternative compiler products. This makes the PathScale EKO

Compiler Suite the clear choice for AMD Opteron computing clusters focused on maximum performance.

LOW-LEVEL AMD OPTERON-SPECIFIC OPTIMIZATION

The PathScale EKO Compiler Suite maximizes the performance of application code when compiled for AMD Opteron systems. With AMD Opteron advantages like complex addressing modes, large register sets, more efficient parameter passing, and SSE2 support, 64-bit code will simply perform better on this processor. In addition, inline AMD64 assembly code can also be issued.

AVAILABLE AS AN ANNUAL SUBSCRIPTION SERVICE

The PathScale EKO Compiler Suite is sold as a subscription service, which includes all major and minor product enhancements, bug fixes, performance tuning, and access to customer support. The subscription is sold on a per developer basis.

MATURE, SUPPORTED TOOLSET

PathScale's Compiler Suite leverages the Open64 and GNU compilers. PathScale provides advanced technical support, and rapid bug response by some of the best engineering talent in the industry.



PATHSCALE™ COMPILER SUITE BENEFITS

Binary and Source Code Compatibility

- No source code changes are required; uses GNU gcc/g++ compatible language
- Fortran 77/90/95 with Cray/SGI Fortran 95 extensions, upward compatible from Fortran 77
- Standard compiler flags are used, so most makefiles generally operate without modification
- Can mix and match GCC and PathScale-compiled objects

Linux System Compatibility

- Available as a Linux installable RPM or tar file
- Uses standard GNU toolchain, including ld, as, and gdb

PATHSCALE COMPILER ENGINEERING

The PathScale Compiler engineering team is led by Dr. Fred Chow. Dr. Chow was Chief Scientist at SGI for the MIPS-based compilers and the lead architect of the Pro64 compiler. Before SGI, Fred was the lead compiler architect at MIPS Computer Systems. The MIPS compiler was a productization of Fred's pioneering Ph.D. research at Stanford. Fred currently holds eight patents.

PATHSCALE™ EKO COMPILER SUITE FOR AMD OPTERON

PATHSCALE EKO COMPILER SUITE FEATURES

Functional Components

- GNU C front-end compatible with gcc 3.3.1
- GNU C++ front-end compatible with g++ 3.3.1
- Fortran 95 with Cray/SGI and Fortran 95 extensions
- Compiler drivers compatible with GNU
- Fortran Libraries

MAJOR OPTIMIZATIONS

Code Generation

- Control flow optimization
- If-conversion
- Instruction scheduling
- Global register allocation
- Loop unrolling
- Peephole optimization

LOOP NEST OPTIMIZER

- Loop-caused dependency analysis
- Loop Interchange
- Cache Blocking
- Loop Fission
- Loop Fusion
- Outer loop unrolling
- Prefetching
- Scalar Expansion and Array Expansion
- Gather-scatter
- Pad arrays to reduce cache conflicts
- Vectorization (including SIMD)

INTER-PROCEDURAL ANALYSIS

- Operates across multiple compilation runs
- Function inlining
- Inter-procedural constant propagation
 - Parameters and global variables
 - Function Cloning
- Dead function elimination
- Dead variable elimination
- Automatic common block padding

The PathScale EKO Compiler Suite contains on a common distribution medium both PathScale proprietary and open-source software components. All open source software is licensed in full compliance with the applicable open source license. PathScale modifications to GPLed source are licensed under the GPL. End-users can receive source code for all of the open source components of the PathScale EKO Compiler Suite, along with their binary distribution, in compliance with section (3a) of the GPL. Open source license terms do not apply to PathScale's proprietary code, which includes code such as run-time performance-enhancing libraries that are independent and separate works. End-users may not transfer, sublicense or distribute PathScale proprietary code to any third party. Except as permitted in the PathScale Subscription Agreement, end-users may not copy or modify PathScale proprietary code.



BINARY AND SOURCE CODE COMPATIBILITY

C/C++

- Source compatible at GNU 3.3.1 (or later)
- 100% binary interoperability, including name mangling
- Mix and match GNU and PathScale compiled libraries and objects

Fortran compatibility

- Source compatible Fortran 77/90/95 with Cray/SGI 95 extensions
- All common Fortran 77 extensions for compiling "dusty deck" code
- Custom libraries included, including a libm compatible library

AMD OPTERON SPECIFIC OPTIMIZATION

- Support for inline assembly code
- Long address support
- Supports AMD64 ABI (x86-64)
- Utilizes full 64-bit ISA including SSE/SSE2 and large register sets
- Instruction Scheduling for AMD's complex Out-of-Order core

GLOBAL SCALAR OPTIMIZATION

Pre-optimizer

- Goto conversion
- Loop normalization
- Alias analysis (flow-free and flow-sensitive)
- Tail recursion elimination
- Dead store elimination
- Induction variable canonicalization
- Copy propagation
- Dead code elimination

Main optimizer

- Partial redundancy elimination based on SSAPRE framework
 - Global common sub-expression
 - Loop invariant code motion
 - Strength reduction
 - Linear function test replacement
- Induction variable elimination
- Register promotion

Tested on the following Operating Systems:

SuSE Linux Professional 9
SuSE Linux Enterprise Server 8 for AMD Opteron
Red Hat Enterprise Linux Workstation 3
Red Hat Fedora Core1

PATHSCALE, INC. TEL 408.746.9100
477 NORTH MATHILDA AVENUE FAX 408.746.9150
SUNNYVALE CA 94085 USA PATHSCALE.COM