DSS makes Floating Point Libraries Fly with Solaris Threads

CASE STUDY

Over 10,000 of the most widely used engineering and scientific applications can now take full advantage of the speed of multiple processor SPARC® computers by simply relinking their existing applications. Dakota Scientific Software (DSS) is making computers much faster and easier to use for engineering and scientific users with the release of DSSLIB, an optimized implementation of the standard BLAS, LAPACK, and LINPACK libraries that use Solaris™ threads. There are literally tens of thousands of programs built with these libraries including such well-known favorites as IMSL/Math from Visual Numerics, LAPACK.h++ from Rogue Wave, and NASA's NASTRAN. All of them can start running 2-20 times faster within 10 minutes simply by relinking with DSSLIB. Even users of single processor machines can often enjoy speedups of 40% or more with DSSLIB.

Dakota Scientific Software was founded in 1990 when several employees from a Minneapolis supercomputer company decided that they had a better way. They realized that many of the techniques that they were using on their company's $35 million supercomputers could also be used to significantly increase the power of more cost-effective RISC platforms. "Our customers are some of the most demanding engineering and scientific customers in the world with large-scale computing needs that require the best possible performance from their Sun computers," says Diane Boucher, DSS President.

Performance is a constant problem with engineering and scientific applications. Applications for structural analysis, CFD, VLSI design, and reservoir modelling are among the most computationally demanding programs that exist anywhere. Historically these large applications ran only on specialized supercomputers, but people today are finding that powerful RISC machines can handle the job. "Our customers use computing environments that span the spectrum from a single-processor IPX™ to a large SPARCserver™ 1000 machine, but they all need the maximum speed possible in order to run their applications. We need to deliver that speed across the total Sun product line in order to satisfy our customer base."

Solid Tools Support

For DSS, SunSoft was the natural choice to supply the integrated suite of tools required to manage the ongoing development of a large software project. "We started by using the performance measurement tool to see where we should apply additional optimization by hand. After optimization, we decided to experiment with the automatic parallelization. Once we saw how well SunSoft's compiler automatically parallelized our optimized code, I knew that we had found our compiler vendor," declares Gail Schmidt, one of the technical team leaders. The level of integration and interoperability among the SunSoft tools has also been an important asset. "Our product comprises around one million lines of code written in a mixture of FORTRAN, C, and assembly. We cannot use a debugger that works well only with our C code, nor are we willing to learn a different source browser for each language." (The SunSoft debugger automatically adjusts itself to the source language.)

After considering many of the alternatives available on the SPARC platform, DSS has chosen to use SunSoft tools in all aspects of their development process. "There are DSSLIB users on Solaris 1.x and 2.x, single-processor and MP, all of the f77 releases with and without iMPact™, and on all types of SPARC processors," noted Schmidt. "We also need to maintain special configurations for some of our licensees. Only SunSoft has the speed, flexibility, and superior support that we need for all of the environments that are important to our customers. We have been very pleased with the quality and selection of SunSoft offerings, and we expect to continue using them to meet our future development needs."

Combining Flexibility with Performance

The flexibility of the Solaris threads interface allowed DSS to implement two models of parallel computation within DSSLIB to meet two distinct needs. SunSoft's FORTRAN MP model of parallelism is very CPU-intensive but gives the highest possible performance. DSS used direct calls to the threads library to implement a second model of parallelism that allows users to trade away a small amount of speed to gain additional CPU resources for other processes. "We were very impressed with the flexibility of the threads interface. None of the other parallel systems that we considered allowed us to implement the two models of parallelism that our customers need," says Schmidt.

In addition to flexibility, the threads delivered the efficiency and high speed that DSS customers demand. "Other parallelization schemes gave us only about 70% CPU efficiency on large problems and they simply were not efficient enough to work at all on medium- or small-sized problems. With Solaris threads, we get efficiency ratings well over 90% on large problems and we are able to parallelize problems of almost any size." Although DSS's customers are not concerned with the details of threads versus other methods, they certainly like the results. "DSSLIB has tripled the speed of the neural networks that we use in our work with satellite remote sensing," says Dr. Toni Logar, a noted researcher at the Institute for Atmospheric Sciences.

SunSoft's Compilers and Tools

SunSoft's Developer Tools division produces a set of compilers and tools to assist in the writing, debugging, and tuning of MT programs, known as SPARCworks™ /iMPact.

SPARCompiler™ FORTRAN MP is an industry standard f77 compiler which can examine a standard FORTRAN program, and automatically create multiple threads to run independent sections of code simultaneously. It detects loops that can be parallelized and generates code that can execute on multiple processors. The number of processors to be used is determined by the user at run time. For loops which are not amenable to the analysis, the user can insert explicit directives to tell the compiler that they can be parallelized. The compiler will also produce output indicating which loops were parallelized, and why other loops couldn't be.

SPARCworks Debugger MT extends the debugger to support dynamic analysis and control of threaded programs. It will debug C, C++, and FORTRAN MT applications, switching between the various languages as necessary. Setting breakpoints, inspecting data, navigating up and down the stack are all accomplished with respect to the individual thread chosen. The programmer can select the thread of interest with the ThreadInspector, and work with that one alone, or execute commands that will affect all threads.

All of this is in addition to the standard Debugger features which include "Runtime Error Checking", (which checks for memory access violations and leaks), "Fix and Continue" (which allows the programmer to recompile only that code which has changed, and then incrementally link it back into the image being debugged), Watchpoints, a scripting language, a graphical data inspector, and all the other normal debugger functionality.

For more information about how you can take advantage of multithreading, please send email to Threads-Hotline@Sun.COM, or call 415 336-3032.

SunSoft
A Sun Microsystems, Inc. Business

2550 Garcia Avenue, Mountain View, CA 94043-1100 USA ● (415) 960-3200 ● Fax (415) 961-6070 Printed in USA 9/94 94338-001