Pacific Northwest NATIONAL LABORATORY

Proudly Operated by **Battelle** *Since 1965*

"The programming environment on the Cray XMT is quite sophisticated and supports automatic parallelization, as well as automatic multithreaded code generation. However, several important features are missing from the environment, including support for task parallelism, exposing more detailed resource control to the programmer, as well as optimizations enabled by the newer architectural features available on the XMT."

- Pacific Northwest National Laboratory Task Lead Daniel Chavarria

U.S. DEPARTMENT OF ENERGY

Center for Adaptive Supercomputing Software-Multithreaded Architectures (CASS-MT)

Compiler and Runtime System

At a glance

The goal of this research is to extend existing Cray XMT C/C++ compiler and run-time framework to support alternative parallel code structures and exploit XMT architectural features. This involves making several enhancements to the Cray XMT's compiler that result in productivity and performance benefits. These extensions increase Cray compiler effectiveness for many classes of applications by better thread management, locality optimizations and support for dynamic nested parallelism.

What we do

At Pacific Northwest National Laboratory's (PNNL) CASS-MT, the research team is making it a priority to develop techniques and tools to improve the success of various applications used on the XMT. Programming languages and their compilers tell a computer what to do. Programming languages can be as different from each other as Spanish is from English and German. The compiler acts as an interpreter between the human and the machine – translating source code (closer to human language) into the machine's unique language. The Cray XMT uses an extended version of the C/C++ programming language. By improving the translator, you ultimately can improve the speed and capabilities of the machine.

The XMT inter-node communication architecture is based on the XT-3's interconnect, which is designed for distributed memory systems and applications that exploit locality. They team will research several enhancements to the MTA programming model and compilation process. Research areas include: Compiler thread generation and destruction exposed to the programmer; Result reproducibility and/or accuracy of transient reductions; Mechanisms for exploiting locality to improve scalability, as well as support for more Advanced parallelization strategies (such as dynamic nested parallelism).

How we do it

The current programming environment on the XMT focuses on data-parallel compilation of loops nests. That is the threads on the XMT system execute the same loop bodies over different portions of the data. This approach enables the utilization of multiprocessor, multithreaded parallelism across large amounts of data. The compiler and runtime systems in cooperation, determine when to create and destroy threads and how to map loop iterations to threads. However, for some application areas this is not enough and more detailed control of the parallelism is needed.

CASS-MT's team will make these capabilities available to users of the XMT to expand and improve existing and future applications.

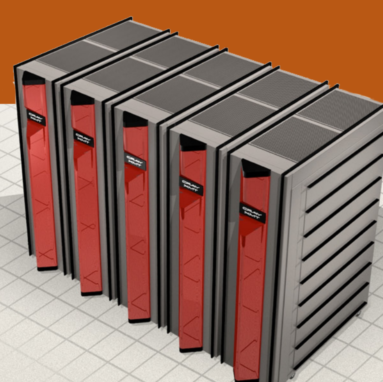
CASS-MT is dedicated to research on systems software, programming environments, and applications in a High-Performance Computing (HPC) multithreaded architecture environment.

We offer the only Open Science Cray XMT system, a one-of-a-kind supercomputer consisting of 128 multithreaded processors, 1 TB RAM, and a 7.7 TB Lustre parallel filesystem.

The Cray XMT supercomputer has the potential to substantially accelerate data analysis and predictive analytics beyond the limitations of traditional computing. Multithreaded processors allow multiple, simultaneous processing, helping researchers find solutions to the world's most complex challenges faster. The XMT can process irregular, data-intensive applications that have random memory access patterns. Unlike many applications where data delivery is dependent on memory speed, the Cray XMT's multi-threaded architecture tolerates memory access latencies by switching context between multiple threads that work continuously, overlapping the memory latency and preventing the processor from being held up while it waits for data to arrive.

The multithreaded technology powering our Cray XMT is ideally suited to perform pattern matching, scenario development, behavioral prediction, anomaly identification, and graph analysis.

Try it for yourself. We seek to create collaborations and provide expertise for porting and optimizing applications. The opportunity to use our Cray XMT system is available to internal and external research partners.



John Feo,
CASS-MT Director
(509) 375-3768
John.feo@pnl.gov
cass-mt.pnl.gov/

Daniel Chavarria
Task Lead, High-Performance Computing
509-372-6964
daniel.chavarria@pnl.gov

Pacific Northwest
NATIONAL LABORATORY

Proudly Operated by Battelle *Since 1965*