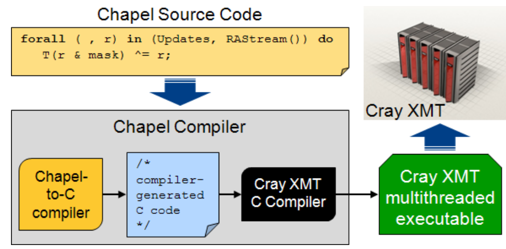
Center for Adaptive Supercomputing Software-Multithreaded Architectures (CASS-MT)

Chapel for the Cray XMT

Pacific Northwest NATIONAL LABORATORY

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

At a glance



Chapel is a new parallel programming language being developed by Cray Inc. with the goal of increasing the productivity of the end user. One of Chapel's themes is to support general parallel programming by having the user express parallelism and locality in an architecture-neutral manner using high-level abstractions. For programmers of multithreaded architectures like the Cray XMT, this has great promise since dominant HPC programming models like MPI are a poor fit for it while its native programming model does not support parallel execution on other architectures. To help fulfill this promise, this project is focused on improving Chapel's level of support for the Cray XMT.

"The Cray XMT's unique architectural features for multithreading have also resulted in a unique programming style since most current HPC programming models, like MPI, expose too many assumptions about the target architecture in the user's code. In our work with Chapel, we strive to let the programmer focus on the abstract concepts of parallelism and locality, permitting programs to target a more diverse set of platforms including desktop multicore architectures, traditional distributed memory systems, and now through the CASS-MT program, multithreaded architectures like the Cray XMT."

- Cray Inc Task Lead
Brad Chamberlain

What we do

The Chapel compiler being developed by Cray researchers as part of the DARPA High Productivity Computing Systems program uses source-to-source compilation to implement a user's Chapel program via standard C code with calls to runtime libraries that implement the necessary parallelism and communication. This permits the Chapel compiler to portably target such diverse architectures as multicore desktops, commodity clusters, and Cray supercomputers (not to mention those developed by other vendors).

Chapel's support for the Cray XMT has traditionally lagged behind other architectures due to the fact that within HPCS, Chapel has focused primarily on supporting large-scale distributed memory systems and only on multithreading only at a small scale.

As part of PNNL's Center for Adaptive Supercomputing Software-Multithreaded Architectures (CASS-MT) the Cray research team is modifying the open-source Chapel compiler so that its generated C code can automatically be parallelized by the standard XMT C compiler when it serves as the back-end compiler. This will permit standard data parallel constructs in Chapel to transparently make effective use of the thousands of hardware thread contexts supported by the Cray XMT. The team will then implement XMT-specific performance optimizations with the goal of making Chapel's performance competitive with user-written native XMT C.

In addition to improving the Chapel compiler to make more effective use of the Cray XMT, this project also focuses on extending the language and compiler to permit a single Chapel program to execute in parallel across a variety of distinct architectures. One such example would be to have a Chapel program execute using the compute *and* service nodes of the Cray XMT. A second would be to have a single program execute using a Cray XMT in combination with distinct external systems such as a desktop computer, Cray CX1000, and/or Cray XE6.

To this end, a new locality feature—the *realm*—has been added to the Chapel implementation to represent distinct target architectures. This permits Chapel

U.S. DEPARTMENT OF ENERGY

programmers to specify the node types that should be used for each sub-computation within their program. As an example, a user could specify that a large unstructured graph computation should execute on the realm representing a Cray XMT's compute nodes while a distinct part of the program requiring dense linear algebra could execute simultaneously using the XMT's service nodes or the compute nodes of a more traditional external system.

How we do it

To improve the mapping of Chapel programs to the Cray XMT, the Chapel team has been modifying the Chapel compiler and its standard modules to generate C loops and XMT-specific pragmas in order to generate the parallelism required to make effective use of the Cray XMT. In addition, the team has been implementing new scalar optimizations in order to reduce memory traffic and generate performance competitive with user-written C for the Cray XMT. To support the realm concept, the Chapel compiler and runtime libraries have been improved to better support distinct target architectures, potentially with different native data sizes and formats.

Future Applications

- Further tuning of the Chapel compiler to better support the Cray XMT
- Using the Lightweight User Communication Environment (being developed by a distinct research team within CASS-MT) to support the realm concept for the Cray XMT
- Exploring the applicability and benefits of Chapel and realms for the application areas being researched by other teams within CASS-MT

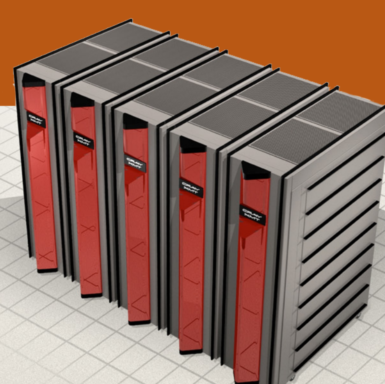
CASS-MT is dedicated to research on systems software, programming environments, and applications in a High-Performance Computing (HPC) multithreaded architecture environment.

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/

Brad Chamberlain
Task Lead
Cray Inc.
bradc@cray.com

THE SUPERCOMPUTER COMPANY

Pacific Northwest
NATIONAL LABORATORY

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

PNNL-SA-74689