

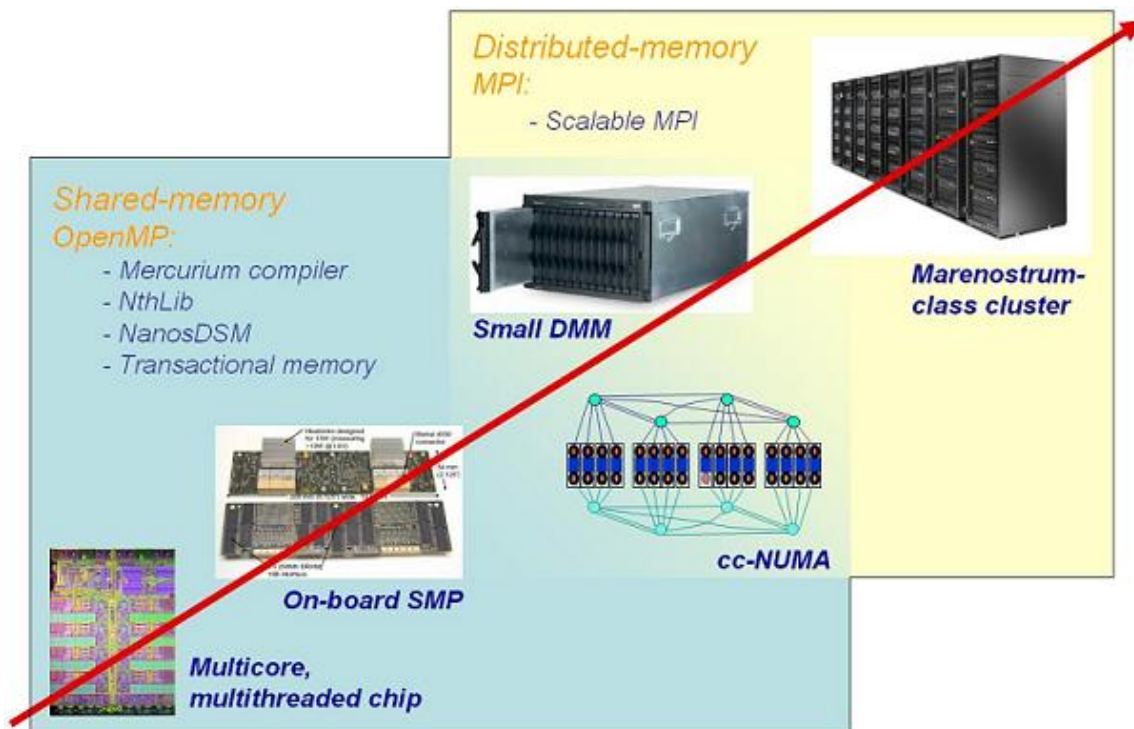
## Parallel Programming Models

### OVERVIEW

C/C++ and Fortran are widely used programming languages for sequential applications. There is a variety of parallel programming models based on them. MPI is build like a set of message passing primitives, usually called from C and Fortran. OpenMP proposes extensions to such languages. There are other programming languages that extend them in different ways: Unified Parallel C and Co-Array Fortran. And there are new language proposals like Fortress (SUN), Chapel (Cray), and X-10 (IBM), which promise improved programmer productivity. Our team explores such programming environments and provides support for the execution of parallel applications on top of some of them.

### OBJECTIVES

The main goal of the team is to investigate new and current programming paradigms and the associated runtime system support to provide high performance to parallel applications. The target architectures range from multicore and SMT processors to shared- and distributed-memory systems. The Cell BE processor is currently one of our main targets.



- At the programming model, we are proposing extensions to OpenMP to improve the expressiveness of the model. The Mercurium Compiler is able to exploit multiple levels of parallelism and generate work from multiple simultaneously executing threads. Once parallelism is spawned on a coarse level, new opportunities for parallelism on a finer level result in the generation of work for all or for restricted groups of processors. Although our primary focus is on OpenMP, we are also investigating ways to exploit the parallelism in distributed memory architectures, specially now that new systems start offering local memories that must be managed by the software (i.e., in the Cell

processor).

- The runtime system offers the basic services to spawn/join parallelism and synchronize across threads. The NthLib API is simple enough to be used for multithreaded programming on a variety of multiprocessor platforms. The goal is to have a platform easy to modify to incorporate new functionalities, as needed by the new architectures. For instance, currently is being used to experiment with new ways to exploit parallelism using local memories in the Cell processor.

## PROJECTS/AREAS

- **Nanos runtime system:** NthLib is a user-level threads library primarily designed to provide runtime support as the backend of the Mercurium compiler. The main focus is to provide effective support for multiple levels of parallelism. The library not only supports the structured parallelism offered by OpenMP but also supports the execution of parallel tasks in a non-structured way. This is useful for instance to support applications that use the client-server model.
- **Mercurium compiler infrastructure:** The infrastructure provides support for OpenMP 2.5 for Fortran 77/90 and C. A mechanism based on templates allow the researcher to specify program transformations for each element of the OpenMP programming model.
- **NanosDSM:** NanosDSM is the support of Distributed Shared Memory for Nanos. NanosDSM comes as a run-time library that is used to keep consistent the memory used in different nodes of a cluster, in such a way that a parallel application can be run as if it was on a shared-memory architecture.