SAND2014-19477PE



Run Time Systems R&D with the Qthreads Multithreading Library

Dylan Stark, Stephen Olivier
Dept. 1423, Sandia National Laboratories, NM

ATM Lightening talk

SNL/CA

November 5, 2014

Sandia National Laboratories

Exceptional service in the national interest

U.S. DEPARTMENT OF ENERGY
NNSA National Nuclear Security Administration

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Qthreads Philosophy

Sandia National Laboratories

- Qthreads as a vehicle for run time system research
  - Co-design efforts with architecture and applications
  - Modular for flexibility and extensibility
    - Interfaces to OpenMP, Kokkos, Chapel, <your language here>
    - Different schedulers, e.g., work stealing, hierarchical
- Crowded space of run time system solutions
  - Don't claim to have the best, but strive to improve
  - Still many unsolved problems
    - Want to gain understanding
    - Seek collaboration

Qthreads Overview

Sandia National Laboratories

- Programmer exposes application parallelism as massive numbers of lightweight tasks (qthreads)
  - Problem-centric rather than processor-centric decomposition enhances productivity, transparent scaling
  - Both loop-based and task-based parallelism supported
  - Full/empty bit primitives for powerful, lightweight synchronization (emulates Tera/Cray MTA/XMT behavior)
  - C API with no special compiler support required
- Dynamic run time system manages the scheduling of tasks for locality and performance
  - Heavyweight worker pthreads execute the tasks
  - Worker pthreads pinned to underlying hardware

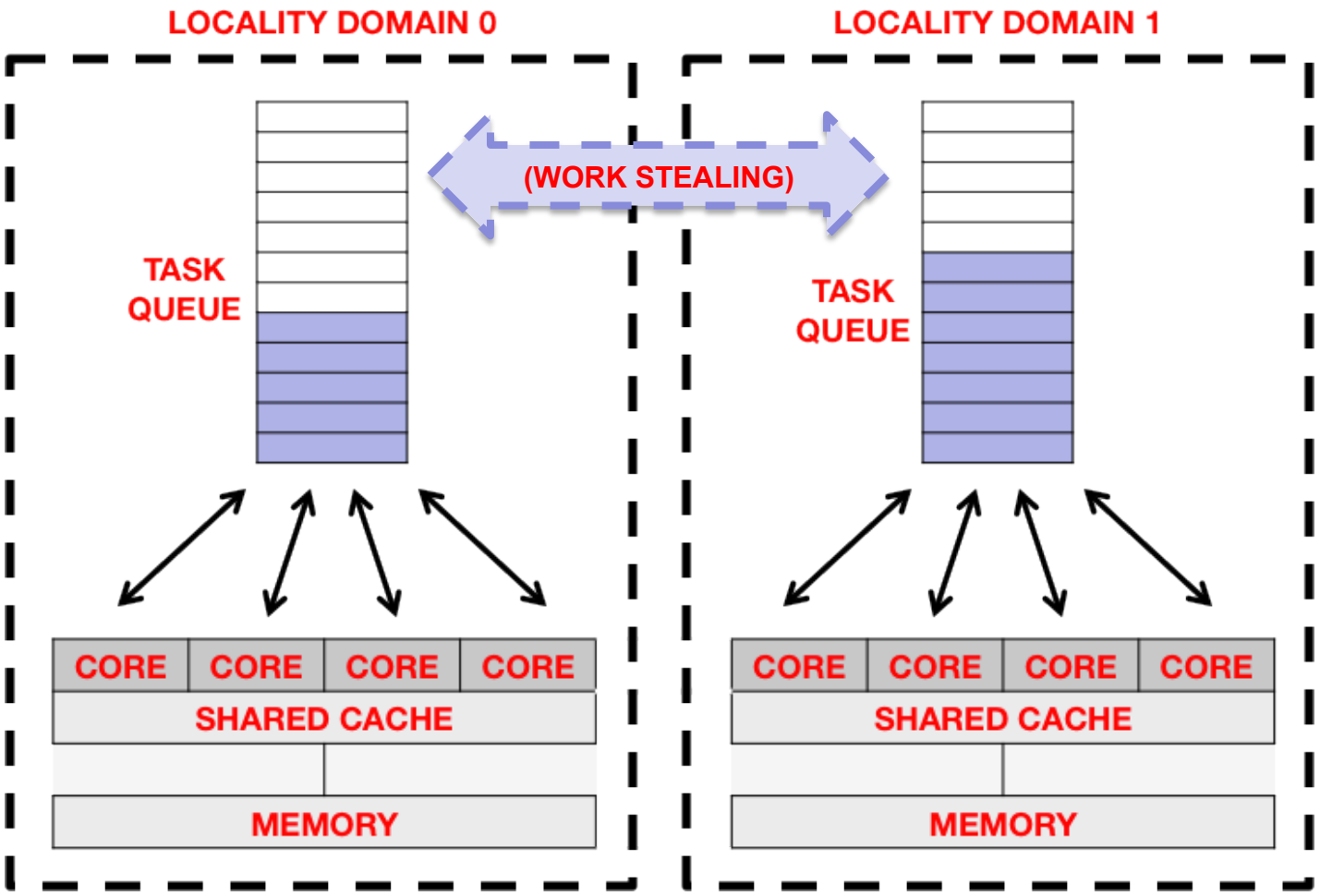
Qthreads Capabilities

Sandia National Laboratories

- Locality-aware load balancing of tasks to support NUMA and complex cache hierarchies
  - Locality domain with work queue shared among worker threads that share cache and memory
  - Work stealing between locality domains for global load balancing
- Lightweight task context switching
- Ported to x86, Phi, PPC, Sparc, Tilera

Qthreads Run Time View of Locality

Sandia National Laboratories



X-over-Qthreads

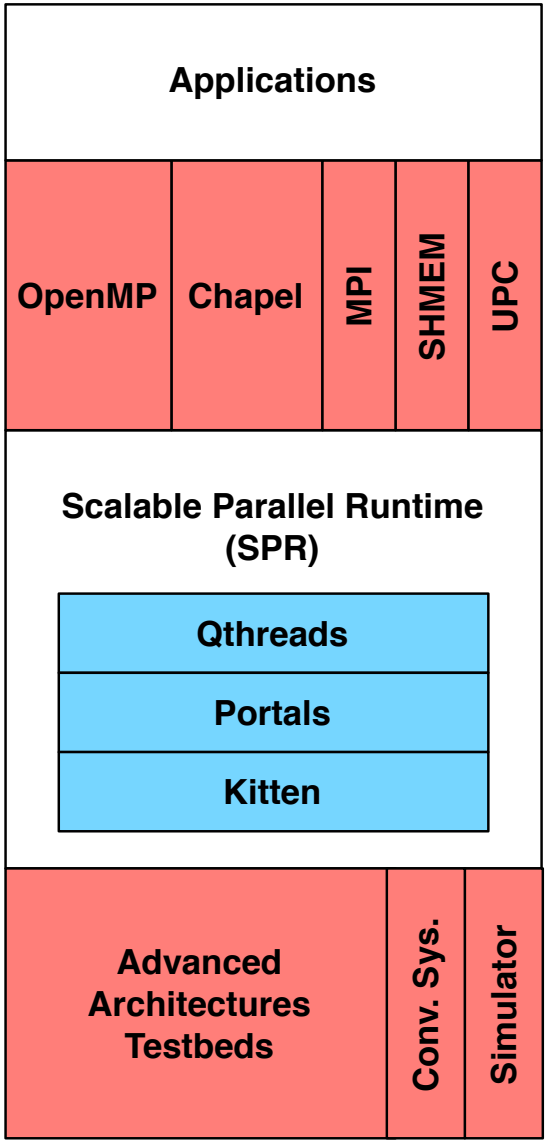
Sandia National Laboratories

- Qthreads as run time for **X**
  - Allows execution of **X** codes without porting to Qthreads API
  - Leverage existing front-ends
  - Enables experimentation with potential new OpenMP features
- **X** = OpenMP (ROSE, Intel)
- **X** = Chapel
- **X** = Kokkos
- And supports composition for MPI+**X**

A Unified Runtime Example

Sandia National Laboratories

- **Qthreads: Lightweight threading interface**
  - Scalable, lightweight scheduling on NUMA platforms
  - Supports a variety of synchronization mechanisms, including full/empty bits and atomic operations
  - Potential for direct hardware mapping
- **Portals 4: Lightweight communication interface**
  - Semantics for supporting both one-sided and tagged message passing
  - Small set of primitives, allows offload from main CPU
  - Supports direct hardware mapping
- **Kitten: Lightweight OS kernel**
  - Builds on lessons from ASCI Red, Cplant, Red Storm
  - Utilizes scalable parts of Linux environment
  - Primarily supports direct hardware mapping



Available Online

Sandia National Laboratories

Qthreads

More info: http://www.cs.sandia.gov/qthreads/

Source: https://code.google.com/p/qthreads/