

# OpenMP 4.5 and Beyond – Webinar part 1 Overview

• Wednesday, June 28<sup>th</sup>, 2017

Scaling OpenMP via LLVM for  
Exascale Performance and Portability (SOLLVE)

PI: Barbara Chapman (BNL)  
Co-PI: Pavan Balaji (ANL)  
Co-PI: David E. Bernholdt (ORNL)  
Co-PI: Sanjay Kale (University of Illinois)  
Co-PI: Vivek Sarkar (Rice University)  
Co-PI: Bronis Supinski (LLNL)

Presenters:  
Tom Scogland  
Oscar Hernandez  
Christopher Earl  
Hal Finkel



# Overview: OpenMP 4.5 Update

- Released at SC'15
- Many refinements to device support
- Clarifications and minor enhancements, including:
  - Reductions for C/C++ arrays
  - Runtime routines to support cancelation and affinity
- Some new features have been added
  - Support for *doacross* loops
  - Divide loop into tasks with `taskloop` construct

# OpenMP 4.5 substantially improves device support

- Unstructured data mapping
- Asynchronous execution
- Scalar variables are firstprivate by default
- Improvements for C/C++ array sections
- Device runtime routines: allocation, copy, etc.
- Clauses to support device pointers
- Ability to map structure elements
- New combined constructs
- New way to map global variables (`link`)

# OpenMP 4.5 has many other refinements to recent additions

- Many clarifications and minor enhancements
  - SIMD extensions
    - SIMD and SIMD parallel loop chunk size control
  - Addition of schedule modifiers: `simd`, `monotonic`, `nonmonotonic`
  - Clarifications of thread affinity policies
  - Grammar for `OMP_PLACES`
  - Support for `if` clause on combined/composite constructs
- Hints for locks and `critical` sections
- Continues to increase Fortran 2003 support
  - Ten limitations remain until 5.0
- Task priorities
- Improved support for C++ reference types
- Compiler support: <http://www.openmp.org/resources/openmp-compilers/>

# OpenMP 5.0 Preview -- TR4 released November 2016

- Major new feature is performance tool support (TR2+)
- Some significant extensions to existing functionality
  - Support for task reductions, including on `taskloop` construct
  - Implicit `declare target` directives and other verbosity reducing changes
- Many clarifications and minor enhancements, including:
  - Use of any C/C++ *lvalue* in `depend` clauses
  - Addition of `depend` clause to `taskwait` construct
  - Addition of `conditional` modifier to `lastprivate` clause
  - Permits `declare target` on C++ classes with virtual members
  - Clarification of `declare target` C++ initializations

# Memory Management --- TR5 released November 2016

- Language features for managing memory on systems with heterogeneous memories.
- Main Concepts:
  - Memory Spaces --- Represents memory resources
  - Memory Traits --- {Location, Distance, Bandwidth, Latency, Persistence, etc}
  - Allocator Traits
  - Allocator and Directives APIs

# OpenMP 5.0 will significantly extend TR4 & TR5

- OpenMP 5.0 is scheduled to be released by SC18
  - TR6 (TBD – SC'17) will document most additions for 5.0
- Main Topics for 5.0
  - Memory locality, affinity and working with complex memory hierarchies
  - Updates to support latest C/C++ standards, completion of Fortran 2003
  - Continued improvements to device support and tasking, including:
    - Deep copy for mapped variables; Improved support for multiple devices
    - Unshackled threads, major extensions for task dependences
  - Interoperability and composability
  - Debugging tools support

# Help us shape the future of OpenMP

- Connect with the SOLLVE project --- WBS# 1.3.1.15
  - Complete our survey on the confluence site!
  - Application engagement via shared milestones for FY17,18,19
- OpenMP continues to grow
  - 28 members currently
- You can contribute to our now planned annual TR or complete specification releases
- Attend IWOMP, become a cOMPunity member
- Become a member in the OpenMP ARB
  - OpenMP membership types will become more accessible
  - Please let us know if you would be interested