SOLLVE OpenMP V&V: The Team

Swaroop Pophale
David E. Bernholdt

Thomas Huber
Nolan Baker
Sunita Chandrasekaran (PI)
Outline

• Introduction
  • What is SOLLVE?
  • Scope and Intent of the Validation and Verification Suite
  • OpenMP Offload (4.x+) Machine Model
  • OpenMP 5.0 New Features
  • OpenMP 5.1 New Features

• Validation and Verification Suite
  • Test design process
  • Infrastructure design

• Result Snapshot
• Success Stories
• Ways to collaborate
Introduction: The SOLLVE ECP Project
Introduction: Scope and Intent of V&V suite

- Verify the status of OpenMP implementations across ECP platforms
- Evaluation of OpenMP functionality and performance on target architectures
- Check implementations’ conformance to the OpenMP standard
- Tests and kernels primarily motivated by ECP Applications
- Verify platforms are ready for OpenMP applications
Introduction: OpenMP Offload (4.x+)

Machine Model

HOST

INTERCONNECT

Device 1
Memory

Device N
Memory

Data and Instructions

Data
New Features OpenMP 5.0

- OpenMP contexts, metadirective, and declare variant
- Addition of requires directive, including support for unified shared memory
- Memory allocators and support for deep memory hierarchies
- Descriptive loop construct
- Release/acquire semantics added to memory model
- First (OMPT) and third (OMPD) party tool support
- Completed support for Fortran 2003
- Added support for Fortran 2008, C11, C++11, C++14 and C++17
New Features in OpenMP 5.1

- Added features:
  - `omp_all_memory` reserved allocator
  - `target_device` trait
  - `target_device` selector
  - `dispatch` construct
  - `assume` directive
  - `nothing` directive
  - device-specific ICV support
  - `error` directive
  - `masked` construct
  - `scope` directive
  - Loop transformation constructs
New Features in OpenMP 5.1 (cont.)

• C++ attribute specifiers support
• Full support for Fortran 2008, C11, C++11, C++14 and C++17 and initial support for Fortran 2018
V&V: Test Design Process

1. **Analyze OpenMP directive OR ECP Application**
2. **Formulate test**
3. **Discuss validity and adherence to specification**
   - **Valid test?**
     - **YES**
     - **NO**
       - **Test passes?**
         - **YES**
           - **Open for review**
             - **Test accepted?**
               - **YES**
               - **NO**
                 - **Bring to OpenMP Specification discussion**
                   - **Specification issue**
                     - **Why?**
                       - **Implementation Bug**
                         - **File Bug report with vendor**
                           - **Add to the V&V suite**
                         - **NO**
                   - **NO**
             - **NO**
         - **NO**
   - **NO**

**Test with available implementations**
V&V: Infrastructure Design

- Our infrastructure is based on a **Makefile + scripts**
- Design parameters:
  - Portability across multiple **compilers** and **systems** and easy to use
    - Support for different compiler options
    - Support for Lua-like Modules and batch schedulers
  - Fast test addition and modification
  - Divided compilation and execution phases
  - Subset of tests selection for partial execution
V&V: Infrastructure Design (cont.)

Test suite

- Tests
  - ompvv
    - 4.5
    - 5.0
    - 5.1
  - declare mapper
    - teams
  - Target teams distribute
  - Results_template
    - Make
    - Scripts
    - Systems
      - Summit.def
      - Cori.def
      - Spock.def

- sys
- template
- Makefile
Sample Results: Current status of OMP Compilers

https://crpl.cis.udel.edu/ompvvsollve/

OPENMP VALIDATION AND VERIFICATION

This website contains all related to the OpenMP Validation and Verification suite developed as part of the Exascale Computing Project (ECP), in particular the Scaling OpenMP via LUM for Exascale Performance and Portability (SCOLUM) project.

This project is a collaboration of:

[Logos of participating institutions]

---

13
Result Snapshot

OPENMP COMPILER STATUS

<table>
<thead>
<tr>
<th>Compiler</th>
<th>Summit</th>
<th>Cori</th>
<th>Spock</th>
</tr>
</thead>
<tbody>
<tr>
<td>XLX</td>
<td>97</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>XLC</td>
<td>143</td>
<td>108</td>
<td>85</td>
</tr>
<tr>
<td>CLANG 13</td>
<td>201</td>
<td>50</td>
<td>39</td>
</tr>
<tr>
<td>GCC 10.2</td>
<td>207</td>
<td>44</td>
<td>37</td>
</tr>
<tr>
<td>CLANG 12</td>
<td>175</td>
<td>75</td>
<td>170</td>
</tr>
<tr>
<td>CRAY 11.0.1</td>
<td>110</td>
<td>140</td>
<td>81</td>
</tr>
<tr>
<td>GCC 10.2.1</td>
<td>210</td>
<td>40</td>
<td>13</td>
</tr>
<tr>
<td>CRAY FTN 9.1.0</td>
<td>92</td>
<td>21</td>
<td>21</td>
</tr>
<tr>
<td>GFORTRAN 10.2.1</td>
<td>100</td>
<td>13</td>
<td>115</td>
</tr>
<tr>
<td>CRAY CC 11.0.4</td>
<td>136</td>
<td>81</td>
<td>170</td>
</tr>
<tr>
<td>CLANG ROCM 4.1.0</td>
<td>170</td>
<td>136</td>
<td>39</td>
</tr>
<tr>
<td>CRAY FTN 11.0.4</td>
<td>85</td>
<td>87</td>
<td>39</td>
</tr>
<tr>
<td>FLANG ROCM 4.1.0</td>
<td>37</td>
<td>37</td>
<td>87</td>
</tr>
</tbody>
</table>
Success Stories

• Cray and AMD are actively using SOLLVE V&V tests to verify correctness and coverage of new features introduced in OpenMP 4.5 and 5.0 (on-going effort).

• Vendors are actively engaging to improve coverage and bug fixes.

• Implementation bugs were identified in the GCC, IBM, and Cray implementations and have been brought to the vendor’s attention.

• OLCF is using SOLLVE V&V as part of acceptance tests for exascale ECP platform Frontier.

• NERSC has integrated SOLLVE V&V for testing OpenMP implementations.
Ways to Collaborate: GitHub

https://github.com/SOLLVE/sollve_vv
Ways to Collaborate: GitHub

• Need to check the behavior of YOUR OpenMP application?
  • Create an application kernel
  • SUBMIT an Issue OR CREATE a PR on https://github.com/SOLLVE/sollve_vv

• Found something we missed?
  • SUBMIT an Issue OR CREATE a PR on https://github.com/SOLLVE/sollve_vv
Ways to Collaborate: Contact information

Swaroop Pophale (pophaless@ornl.gov)
David E. Bernholdt (bernholdtde@ornl.gov)

Thomas Huber (thuber@udel.edu)
Nolan Baker (nolanb@udel.edu)
Sunita Chandrasekaran (schandra@udel.edu)
Acknowledgement

Work supported by the U.S. Department of Energy, Office of Science, the Exascale Computing Project (17-SC-20-SC), a collaborative effort of the U.S. Department of Energy Office of Science and the National Nuclear Security Administration under contract number DE-AC05-00OR22725.
SC’21 Booth Talk Series

For the OpenMP specification, tutorials, forum, reference guides, and links to other resources, visit www.openmp.org