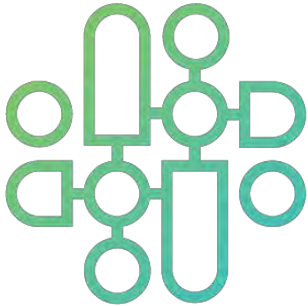


# OpenMP<sup>®</sup>

## SC23 Booth Talk Series



**ECP SOLLVE: OpenMP Validation  
and Verification Testsuite**

**Felipe Cabarcas and Swaroop Pophale**

# ECP SOLLVE: OpenMP Validation and Verification Testsuite

Andrew Kallai, **Felipe Cabarcas**, Sunita Chandrasekaran  
Swaroop Pophale, Seyong Lee, David E. Bernholdt



# SOLLVE V&V Current Team



- Andrew Kallai
- Felipe Cabarcas<sup>1</sup>
- Sunita Chandrasekaran
- Swaroop Pophale
- Seyong Lee
- David E. Bernholdt
- Outside the project contributors
  - AMD
  - Tobias Schuele (Siemens) for his feedback and engagement with the tests
- Former project members
  - Nolan Baker, Michael Carr, Nikhil Rao, Jaydon Reap, Kristina Holsapple, Joshua Hoke Davis, Thomas Huber, Jose M. Monsalve

# SOLLVE V&V Objectives

- Tests implementations of new features introduced in OpenMP 4.5+
- Checks conformance to the specification
  - Peer reviewed, publicly available
- Highlights ambiguities in the OpenMP specification
- Reports status of implementations on primary ECP target platforms
  - Help application and compiler developers
- Exemplifies the use of the new features in OpenMP
- Abstracts application kernels as tests

# Latest OpenMP Specification: 5.2

- Released November 2021
  - Around 27 modifications (new features, deprecated features, behavior changes)
  - Some of these modifications to the specification are implemented in open source compilers
    - GCC status <https://gcc.gnu.org/wiki/openmp>
    - LLVM status <https://clang.llvm.org/docs/OpenMPSupport.html>
  - Leads to writing tests from scratch, with no compilers to test with
- 6.0 specification is scheduled to be released on November 2024

# OpenMP V&V Testsuite repository

- Clone the repo [https://github.com/SOLLVE/sollve\\_vv](https://github.com/SOLLVE/sollve_vv)
- Setup your environment (install or 'module load' your compilers)
- Use our make commands to compile and run a single test or the whole suite
  - Running a single test:

```
make CC=clang CXX=clang++ FC=flang-new OMP_VERSION=5.2 SOURCES=test_name all
```

- Running the entire suite

```
make CC=clang CXX=clang++ FC=flang-new OMP_VERSION=5.2 all
```

# Tests and Coverage

|              | Number of Tests |           |            |            |
|--------------|-----------------|-----------|------------|------------|
|              | C               | C++       | Fortran    | Totals     |
| 4.5          | 116             | 14        | 103        | <b>233</b> |
| 5.0          | 180             | 21        | 126        | <b>318</b> |
| 5.1          | 89              | 4         | 19         | <b>108</b> |
| 5.2          | 15              | 2         | 4          | <b>20</b>  |
| <b>Total</b> | <b>397</b>      | <b>30</b> | <b>252</b> | <b>679</b> |

|     | Approximated coverage of requested features |      |         |
|-----|---|------|---------|
|     | C   | C++  | Fortran |
| 4.5 | 100%  | 100% | 100%    |
| 5.0 | 88%   | 100% | 88%     |
| 5.1 | 87%   | 100% | 40%     |
| 5.2 | 84%   | 100% | 19%     |

- The testsuite not only contains tests for the new features of each specification, but also commonly used combinations by developers
- We have given priority to requested features over complete coverage

# Systems and Compilers Regularly tested

| <i>System</i>                | <i>Perlmutter</i>   | <i>Summit</i>       | <i>Frontier</i>  | <i>Sunspot</i>       |
|------------------------------|---------------------|---------------------|------------------|----------------------|
| <i>Location</i>              | <i>NERSC</i>        | <i>ORNL</i>         | <i>ORNL</i>      | <i>ANL</i>           |
| <i>CPU / GPU</i>             | <i>AMD / NVIDIA</i> | <i>IBM / NVIDIA</i> | <i>AMD / AMD</i> | <i>Intel / Intel</i> |
| GNU (gcc/gfortan)            | YES                 | YES                 | NO               | NO                   |
| LLVM (clang/flang-new)       | YES                 | YES                 | YES              | NO                   |
| AMD ROCM (amdclang/amdflang) | NO                  | NO                  | YES              | NO                   |
| HPE/Cray (cce)               | YES                 | NO                  | YES              | NO                   |
| IBM (xlc, xlf)               | NO                  | YES                 | NO               | NO                   |
| NVIDIA (nvc, nvfortran)      | NO                  | YES                 | YES              | NO                   |
| INTEL (icpx, ifx)            | NO                  | NO                  | NO               | YES                  |



OpenMP Validation & Verification

Search...

- Project
- Publications
- Repository
- Acknowledge and Cite
- Documentation
- Results
- License

This project is part of

Front Page > Results

These results were last reviewed October 16, 2023

Compilers tested on respective systems:  
 Summit - CPU: **IBM POWER9**, GPU: **NVIDIA V100** - GNU (12.2.1), LLVM (18.0.0), XLC(16.01), NVHPC (22.11)  
 Frontier - CPU: **AMD EPYC 7A53**, GPU: **AMD MI250X** - CCE(16.0.6), AMDCLANG (17.0.0), LLVM (18.0.0)  
 Perlmutter - CPU: **AMD Milan EPYC**, GPU: **NVIDIA A100** - GNU (12.1.1), NVHPC (23.1), CCE (16.0.6), LLVM (18.0.0)  
 Sunspot - CPU: **Intel Xeon Sapphire**, GPU: **Intel Ponte Vecchio** - INTEL Compiler (17.0.0)

Filter results:

Search Results

**Compilers**

CC 16.0.6  
 ftn Cray Fortran : 16.0.1  
 cc 16.0.6

**Systems**

frontier  
 perlmutter  
 summit

**OpenMP Specification Version**

4.5  
 5.0  
 5.1

Compiler results  Both  FAIL  PASS

Test run results  Both  FAIL  PASS

**test\_declare\_mapper\_iterator.c**

Path: tests/5.2/declare\_mapper/test\_declare\_mapper\_iterator.c

Compiler: clang 18.0.0

Source code

Compiler result = FAIL

Compiler command: clang -fopenmp -lm -O2 -fopenmp-libs -ffast-math -fopenmp-version=5.2

Compilation time range: Mon 16 Oct 2023 09:30:48 AM EDT

Compilation output:

```

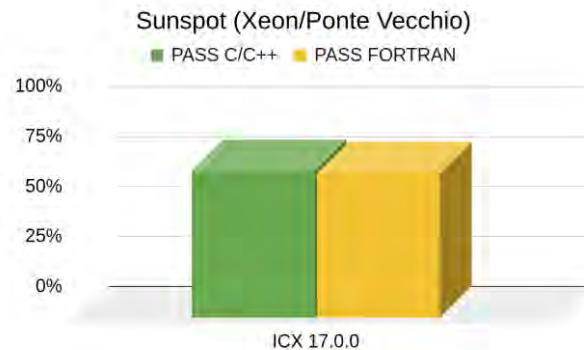
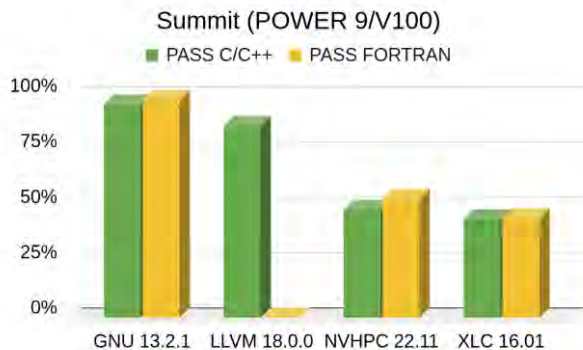
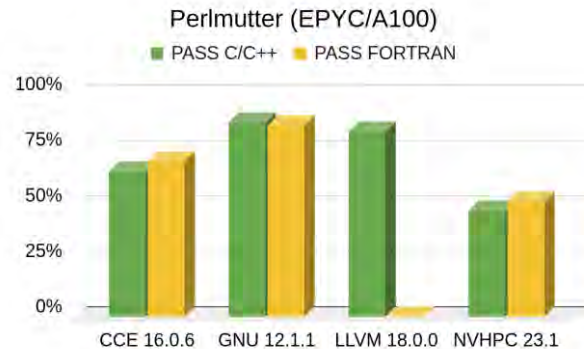
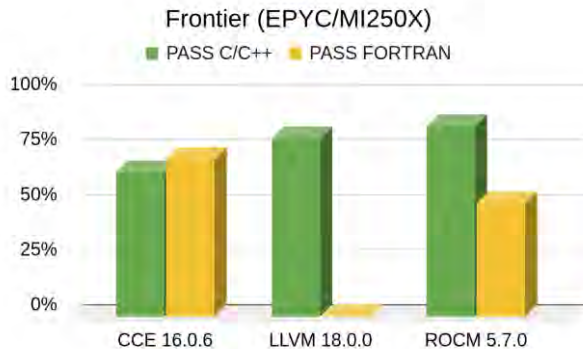
/usr/lib64/gcc/x86_64-suse-linux/7.1.1/_x86_64-suse-linux/bin/ld:
/tmp/test_declare_mapper_iterator-8d1f7fo: In function
`omp_mapper_ZTS5myvec.default':
test_declare_mapper_iterator.c:(text+0x507): undefined reference to `it'
/sw/frontier/ums/ums012/lvrv/18.0.0-20231016/bin/clang-linker-wrapper: error:
`ld' failed
clang: error: linker command failed with exit code 1 (use -v to see invocation)

```

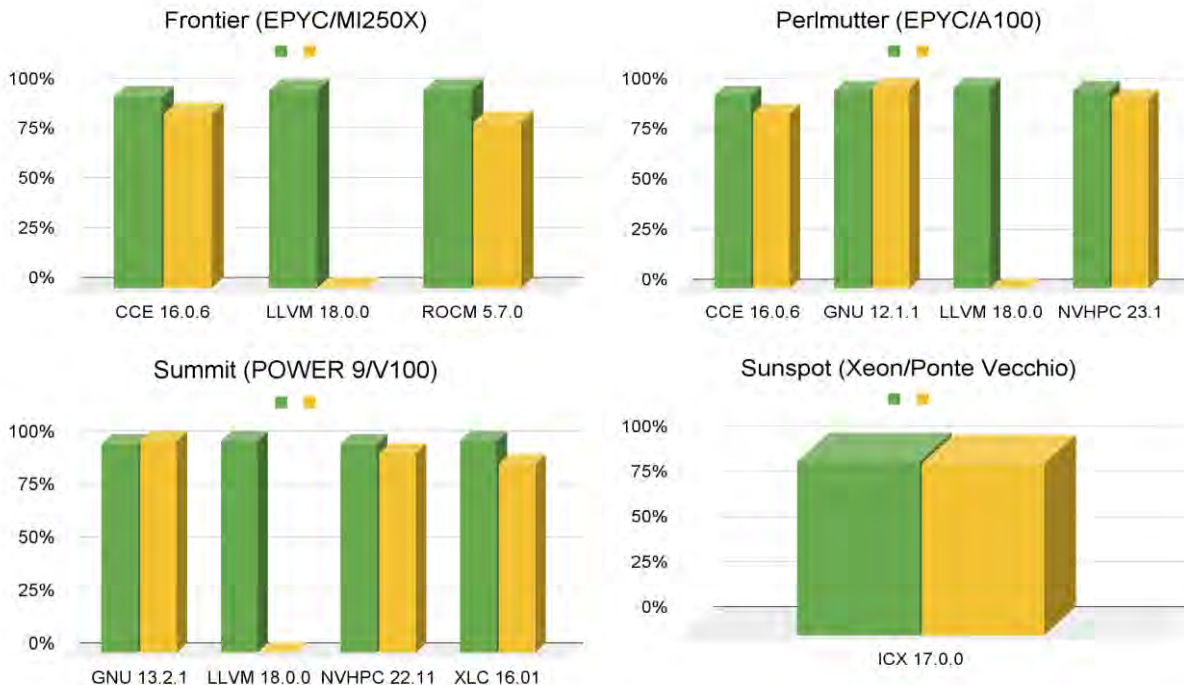
Results Summary table

| # | Source code       | Test name ↓                 | Test system ↓ | Compiler name ↓    | OMP version ↓ | Compiler result ↓ | Runtime result ↓ |
|---|-------------------|-----------------------------|---------------|--------------------|---------------|-------------------|------------------|
| 1 | <a href="#">↗</a> | alpaka_complex_template.cpp | frontier      | CC 16.0.6          | 4.5           | PASS              | PASS             |
| 2 | <a href="#">↗</a> | alpaka_complex_template.cpp | frontier      | clang++ 18.0.0     | 4.5           | PASS              | PASS             |
| 3 | <a href="#">↗</a> | alpaka_complex_template.cpp | frontier      | amdcclang++ 17.0.0 | 4.5           | PASS              | PASS             |
| 4 | <a href="#">↗</a> | alpaka_complex_template.cpp | perlmutter    | CC 16.0.6          | 4.5           | PASS              | PASS             |
| 5 | <a href="#">↗</a> | alpaka_complex_template.cpp | perlmutter    | g++ 12.1.1         | 4.5           | PASS              | PASS             |

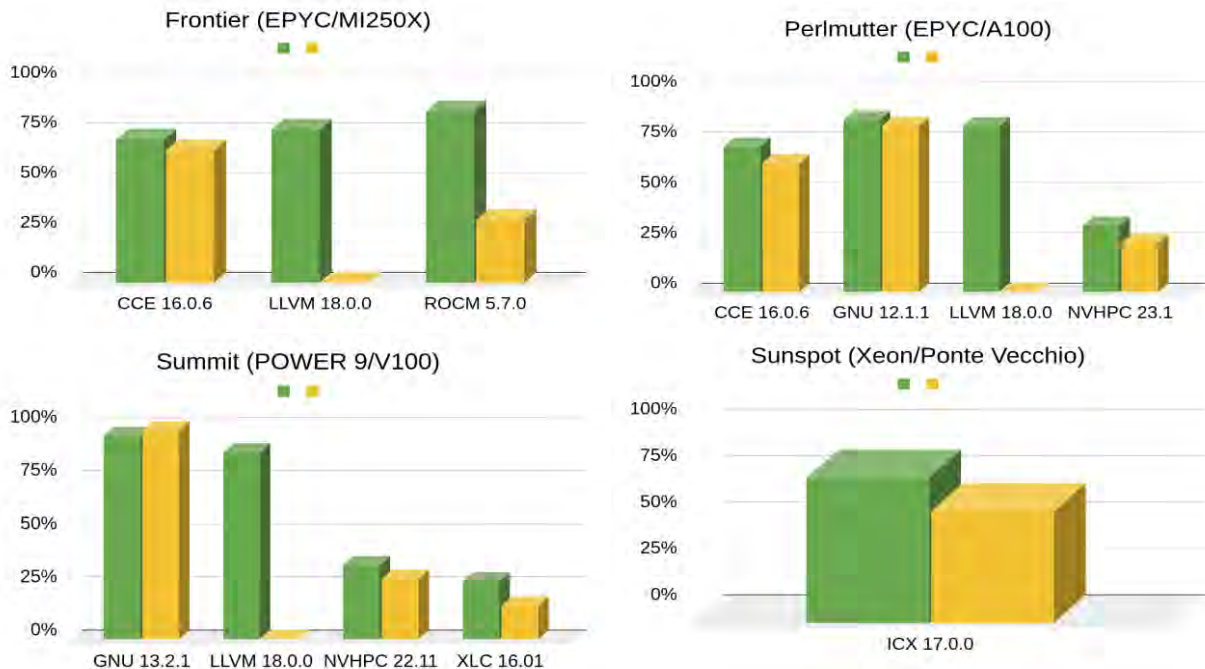
# Overall Results



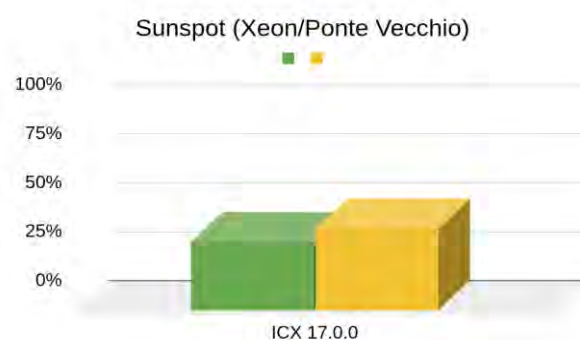
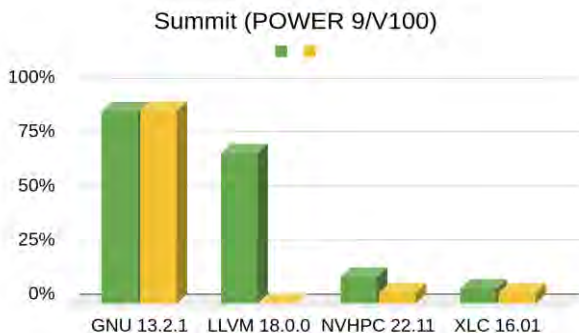
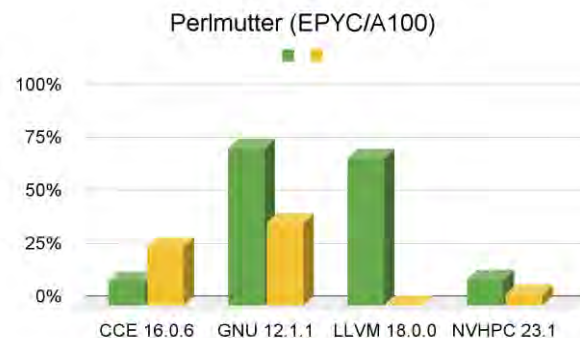
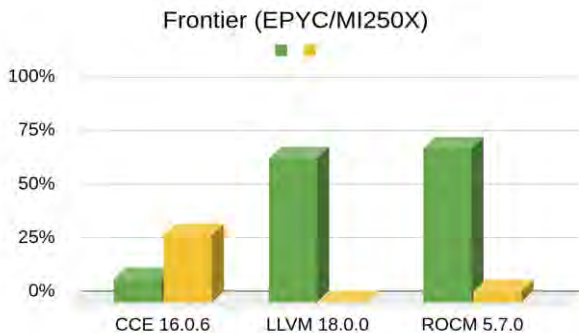
# Results OpenMP 4.5



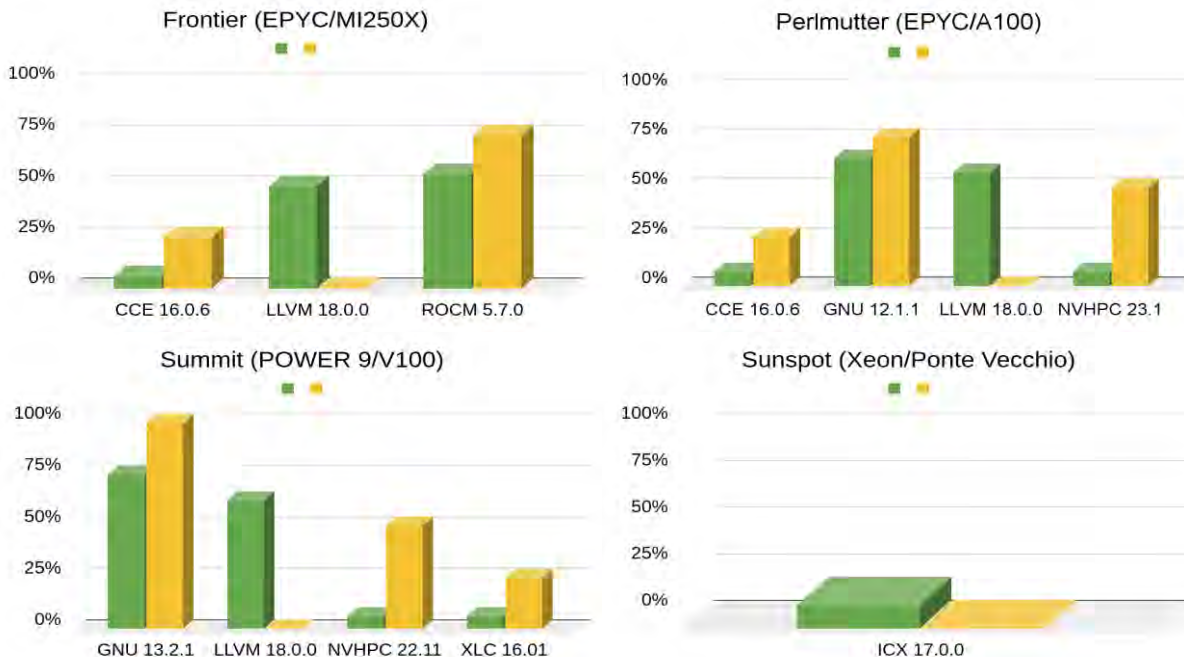
# Results OpenMP 5.0



# Results OpenMP 5.1



# Results OpenMP 5.2



# Beyond V&V testsuite

- At UD: Aaron Jarmusch, Aaron Liu and Vaidhyathan Ravichandran are
  - Integrating regression and stress testing into the SOLLVE V&V Testsuite
  - Creating a CI infrastructure to test LLVM OpenMP on GPU systems
    - Running SOLLVE V&V, Heckbench and AMD's Smoke
    - Focusing on offloading for AMD and NVIDIA GPU systems
      - <https://gitlab.e4s.io/uo-public/llvm-sollve/-/pipelines>
      - Using a system @ University of Oregon
  - Creating an LNT
    - **LLVM Nightly Testing:** a set of client and server tools for monitoring the performance of software over its lifecycle
    - <http://lnt.llvm.org/>

Search GitLab

LLVM SOLLVE

- Project information
- Repository
- Merge requests
- CI/CD
  - Pipelines**
  - Jobs
  - Artifacts
  - Schedules

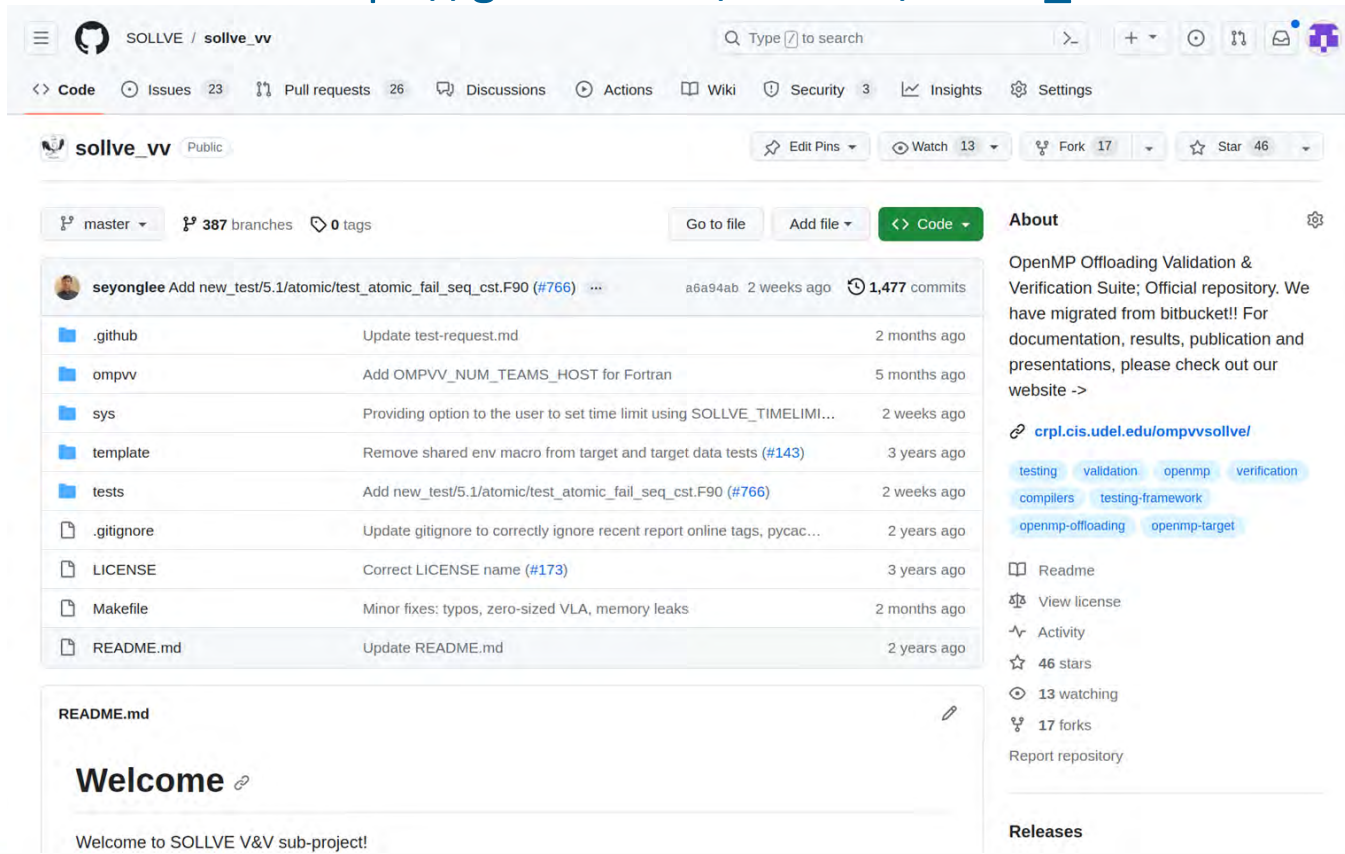
uo-public > LLVM SOLLVE > Pipelines

All 55 Finished Branches Tags

Filter pipelines

| Status                             | Pipeline  | Created by | Stages |
|------------------------------------|---|------------|--------|
| passed<br>00:47:09<br>14 hours ago | Merge branch 'force-build-off' into 'main'<br>#9812 P main → 13d5dc52<br>Scheduled latest |            | ✓      |
| passed<br>00:46:49<br>1 day ago    | Merge branch 'force-build-off' into 'main'<br>#9795 P main → 13d5dc52<br>Scheduled latest |            | ✓      |
| passed<br>00:47:20<br>2 days ago   | Merge branch 'force-build-off' into 'main'<br>#9787 P main → 13d5dc52<br>Scheduled latest |            | ✓      |
| passed<br>00:48:07<br>2 days ago   | Changing the tests that are being run for ...<br>#9785 P gpu-change → 037a3e8b<br>latest  |            | ✓      |
| passed<br>00:54:18<br>3 days ago   | Merge branch 'force-build-off' into 'main'<br>#9779 P main → 13d5dc52<br>latest           |            | ✓      |
| passed<br>00:46:58<br>3 days ago   | hecbench, solve vv: checkout specific com...<br>#9763 P main → 92edf5b7<br>Scheduled      |            | ✓      |





The screenshot shows the GitHub repository page for SOLLVE/sollve\_vv. At the top, there is a search bar and navigation links for Code, Issues (23), Pull requests (26), Discussions, Actions, Wiki, Security (3), Insights, and Settings. The repository name 'sollve\_vv' is displayed as 'Public'. Below this, there are buttons for 'Edit Pins', 'Watch' (13), 'Fork' (17), and 'Star' (46). The main content area shows the file tree for the 'master' branch, with 387 branches and 0 tags. A commit by 'seyonglee' is highlighted, titled 'Add new\_test/5.1/atomic/test\_atomic\_fail\_seq\_cst.F90 (#766)'. The file list includes: .github (Update test-request.md, 2 months ago), ompvv (Add OMPVV\_NUM\_TEAMS\_HOST for Fortran, 5 months ago), sys (Providing option to the user to set time limit using SOLLVE\_TIMELIMI..., 2 weeks ago), template (Remove shared env macro from target and target data tests (#143), 3 years ago), tests (Add new\_test/5.1/atomic/test\_atomic\_fail\_seq\_cst.F90 (#766), 2 weeks ago), .gitignore (Update gitignore to correctly ignore recent report online tags, pycac..., 2 years ago), LICENSE (Correct LICENSE name (#173), 3 years ago), Makefile (Minor fixes: typos, zero-sized VLA, memory leaks, 2 months ago), and README.md (Update README.md, 2 years ago). The README.md content is visible, starting with 'Welcome' and 'Welcome to SOLLVE V&V sub-project!'. On the right side, the 'About' section is visible, stating 'OpenMP Offloading Validation & Verification Suite; Official repository. We have migrated from bitbucket!! For documentation, results, publication and presentations, please check out our website ->' and providing a link to 'crpl.cis.udel.edu/ompvvsollve/'. Below this are tags for 'testing', 'validation', 'openmp', 'verification', 'compilers', 'testing-framework', 'openmp-offloading', and 'openmp-target'. Other repository statistics include 'Readme', 'View license', 'Activity', '46 stars', '13 watching', and '17 forks'.



# OpenMP

## SC23 Booth Talk Series

**[openmp.org](https://openmp.org)**

OpenMP API specs, forum,  
reference guides, and more

**[link.openmp.org/sc23](https://link.openmp.org/sc23)**

OpenMP SC23 booth talk  
videos and presentations