Warwick Data Store

A HPC Library for Flexible Data Storage in Multi-Physics Applications

WARWICK
THE UNIVERSITY OF WARWICK

Objectives and Motivation

- A key focus for modern/future architectures is memory, as raw data read and write speeds lag behind floating-point execution rates.
- Performance-portable applications must manage memory layout to achieve best performance.
- Goal is to design and implement a library that allows for easy tuning of data layouts, without the need for major code rewrites.
- Warwick Data Store (WDS) is a *lightweight*, *performant*, and *extensible* C++ template library that aims to provide data layout abstractions for use in multi-physics applications.

Related Work

- Kokkos [1] and RAJA [2] are designed to facilitate development of performance-portable applications. We focus on data layouts, and make converting between different layouts a first class feature.
- Atlas [3] is designed primarily to support climate and weather applications. WDS aims to support a wider range of different simulation domains.
- Axom [4] is a collection of software packages supporting HPC applications, including Sidre: a data store. Sidre focusses on data description and allocation rather than layout manipulation.
- Intel SIMD Data Layout Templates (SDLT) [5] and bSIMD [6] are abstract data layout libraries aiming to improve auto-vectorisation and enable more effective utilisation of memory bandwidth. WDS looks to provide more flexibility for domain-specific layouts.

Design and Implementation

- WDS has been designed to enforce a clear separation between data description and access.

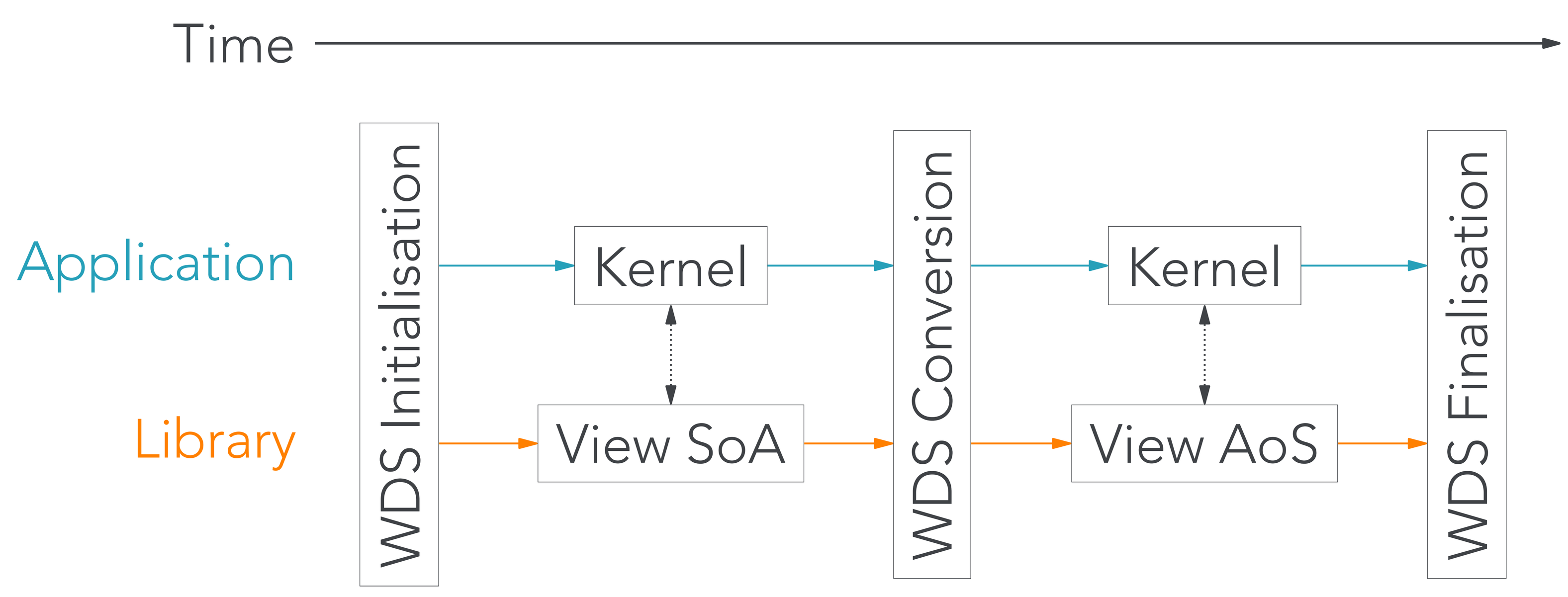

Figure 1: Flow diagram of how WDS can be used. Application kernels are written independently of data layout.

Future Work

- Expand the range and flexibility of data structures the user can call upon. In particular, we plan to fully integrate WDS into a multi-material version of BookLeaf.
- Optimise the conversion process between data structures.
- Implement functionality for migrating data between different memory spaces (NUMA regions, HBM etc.).

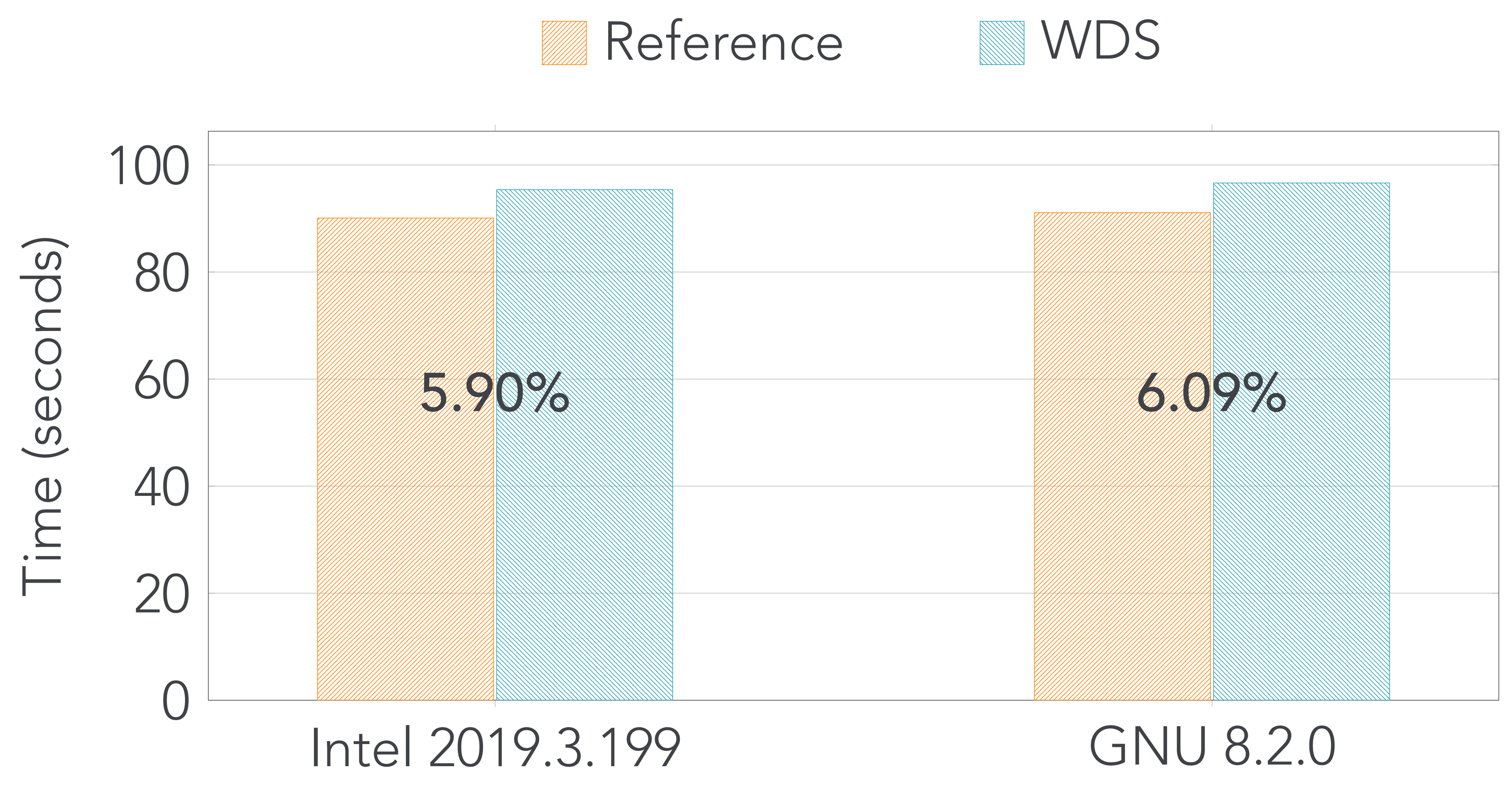

Figure 2: Runtime of BookLeaf [7] with the Sod Lagrangian input deck with a mesh size of 2000×500, utilising 28 OpenMP threads.

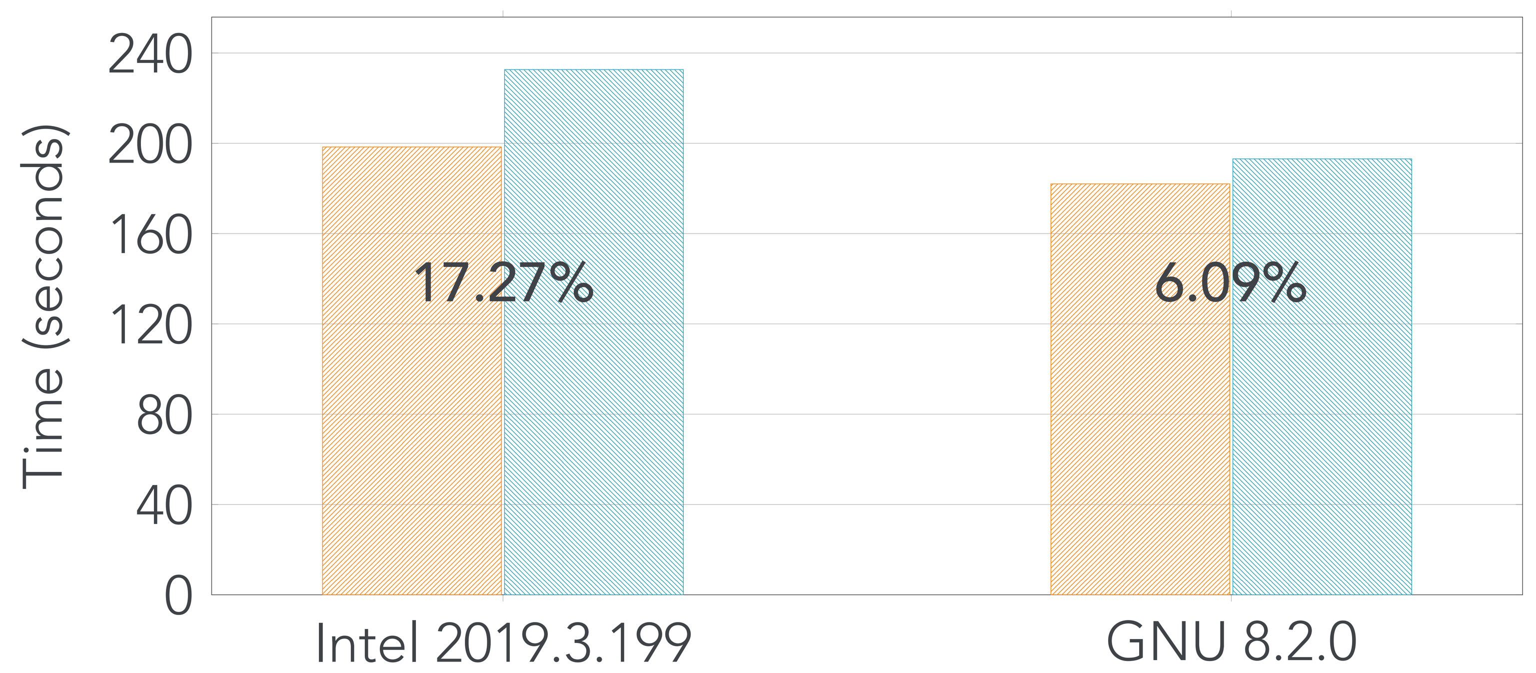

Figure 3: Runtime of MiniMD [8] with a Lennard-Jones potential, utilising 28 MPI ranks with 128^3 particles.

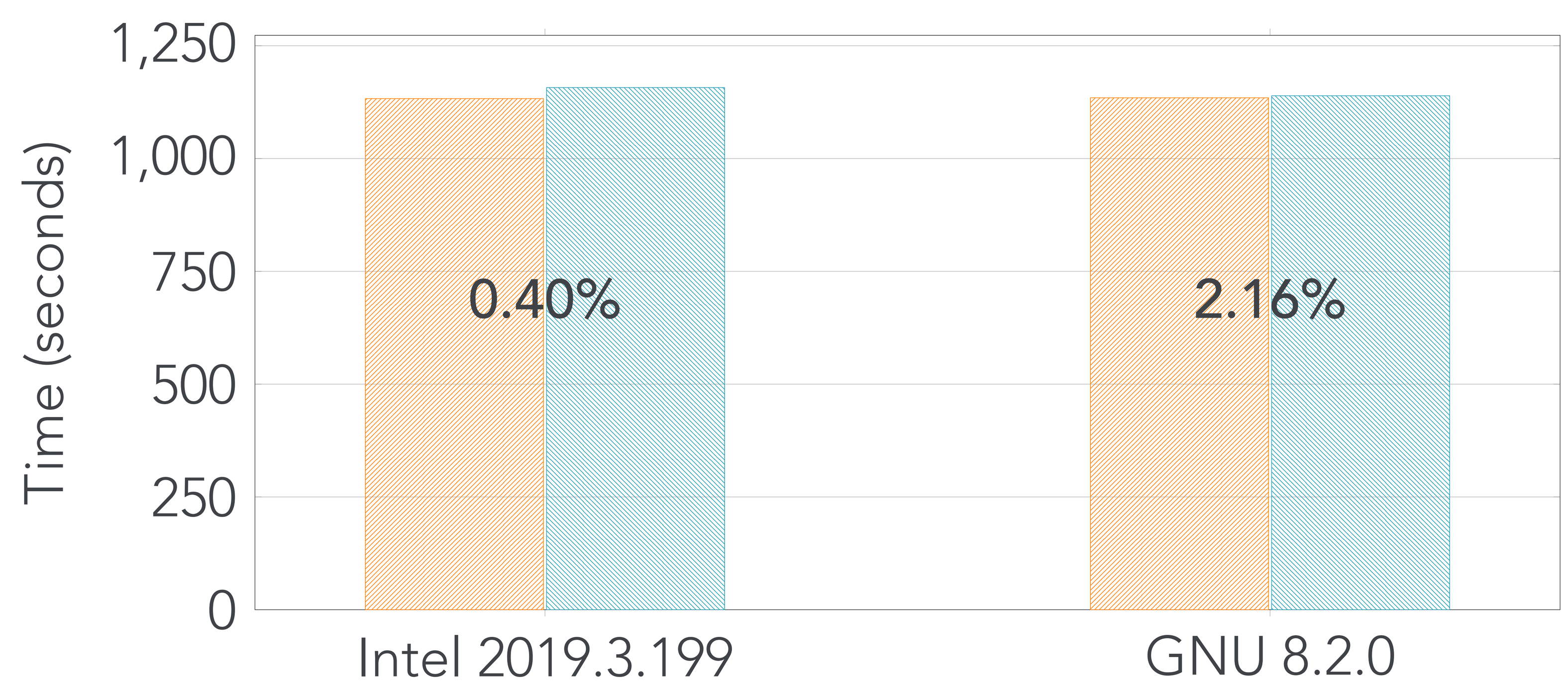

Figure 4: Results from TeaLeaf's [9] Conjugate Gradient solver utilising 28 MPI ranks, with a mesh size of 4000^2.

References

[1] H. C. Edwards and C. R. Trott, "Kokkos: Enabling Performance Portability Across Manycore Architectures," in *Extreme Scaling Workshop (XSW'13)*. Boulder, CO: IEEE Computer Society, Los Alamitos, CA, August 2013, pp. 18–24.

[2] R. D. Hornung and J. A. Keasler, "The RAJA Portability Layer: Overview and Status," Lawrence Livermore National Laboratory, Livermore, CA, Tech. Rep. LLNL-TR–661403, 2014.

[3] W. Deconinck, P. Bauer, M. Diamantakis, M. Hamrud, C. K�hnlein, P. Maciel, G. Mengaldo, T. Quintino, B. Raoult, P. K. Smolarkiewicz, and N. P. Wedi, "Atlas : A library for numerical weather prediction and climate modelling," *Computer Physics Communications*, vol. 220, pp. 188 – 204, 2017.

[4] LLNL, "Axom," https://github.com/LLNL/axom (Accessed: 26^{th} July 2019), 2019.

[5] Jeffers, J. and Reinders, J. and Sodani, A., *Intel Xeon Phi Processor High Performance Programming: Knights Landing Edition*. Elsevier Science, 2016, pp. 251–267.

[6] Numscale, "bSIMD," https://developer.numscale.com/bsimd (Accessed: 7^{th} January 2019), 2017.

[7] D. R. Truby, S. A. Wright, R. Kevis, S. Maheswaran, J. A. Herdman, and S. A. Jarvis, "BookLeaf : an unstructured hydrodynamics mini-application," in *2018 IEEE International Conference on Cluster Computing (CLUSTER)*. IEEE, November 2018, pp. 615–622.

[8] M. A. Heroux, D. W. Doerfler, P. S. Crozier, J. M. Willenbring, H. C. Edwards, A. Williams, M. Rajan, E. R. Keiter, H. K. Thornquist, and R. W. Numrich, "Improving performance via mini-applications," Sandia National Laboratories, Albuquerque, NM, Tech. Rep. SAND2009–5574, 2009.

[9] M. Martineau, S. McIntosh-Smith, M. Boulton, and W. Gaudin, "An evaluation of emerging many-core parallel programming models," in *Proceedings of the 7th International Workshop on Programming Models and Applications for Multicores and Manycores*. New York, NY, USA: ACM, 2016, pp. 1–10.

AWE

Richard Kirk University of Warwick (`R.Kirk@warwick.ac.uk`)
Timothy Law AWE
Satheesh Maheswaran AWE
Stephen Jarvis University of Warwick