

Trials, Tribulations and Joys of Developing with Alternative Parallel Frameworks

Ethan Gutmann*

November 13, 2023

Abstract

Developing large scientific applications is challenging for many reasons, and alternative programming can help with better support for the implementation. These applications need to incorporate the latest domain specific scientific information, be applicable to real world problems, and be robust across a wide variety of inputs. For many models, parallelization is only seen as a necessary hardship, and once developed, the parallelization logic is left untouched, sometimes for decades. Alternative parallel application programming implementations such as Coarray Fortran, Chapel, or UPC++ promise to make implementing and maintaining such parallel logic easier; however, the alternative nature of these implementations often means they do not have the support in the operational HPC community to make that dream a reality. Here we discuss some of the problems that arise implementing the parallelization logic for two different models in Coarray Fortran. We highlight an example from the Intermediate Complexity Atmospheric Research model (ICAR) in which the Partitioned Global Address Space (PGAS) programming model of Coarray Fortran made implementing the generation of a massive lookup-table in parallel almost trivial. We then discuss issues that have arisen since the initial implementation due to inconsistencies in compiler implementations. Improving the support for such parallel frameworks is a bit of a chicken-and-egg problem. Compiler writers do not wish to devote resources to features that are not widely used, and developers do not want to use features without robust support.

*National Center for Atmospheric Research