Debugging SharP: SHARed data-structure centric Programming

Marie McCord – URECA Abstract Spring 2021

Big-Compute and Big-Data applications are increasingly converging, and while system architectures support this convergence there is a lack of supporting programming models, thus creating a software gap. The programming abstraction named SHARed data-structure centric Programming (SharP) aims to address this software gap by abstracting memory systems into a simple and unified user interface that is portable across architectures, providing the user with extensive control over Big-Data organization for Big-Computation. The aim of this study was to debug two key functions related to the distributed hash data structure that were presenting with logical errors. The first function returns the locality information for the data, and the second function moves elements around in the hash data structure in order to create a free slot for new data when one is not available. Both are main features of the program and are critical to its operation. Various debugging tools and techniques were used for this research, which was conducted on a remote hierarchical-heterogeneous computer via Chameleon Cloud. The debugging tool GDB was used to assist in the location of errors, as well as conventional strategies such as backtracking, problem simplification, logging information, assertions, and binary searches. Through this method the bugs were traced respectively through additional functions that were either utilized by or utilize the two functions of original interest, and it was discovered that four other functions required troubleshooting of varying degrees. The goal of this study has been completed as the two functions of interest are now properly working as demonstrated by extensive testing and benchmarking.