Debugging SharP: SHARed data-structure centric Programming

Marie McCord
Sponsor: Dr. Ferrol Aderholdt
Middle Tennessee State University, Computer Science

ABSTRACT

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 SHARed data-structure centric Programming (SharP) abstraction aims to address the software gap by abstracting the memory kinds present on these systems and presenting users with simple, unified interfaces to allocate and manage data in distributed parallel environments.

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 testbed provided by 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 main functions of interest, where 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 testing and benchmarking.

OBJECTIVES

- Debug two key functions related to distributed hash data structure
- Test functions to ensure proper execution
- Benchmark functions and compare to previous results

MATERIALS & METHODS

- A testbed at the Texas Advanced Computing Center (TACC) was provided by a Chameleon Cloud grant
- The tool GNU Debugger (GDB) was used to assist in the location of errors
- Strategies and techniques employed were:
  - Backtracking
  - Problem Simplification
  - Logging Information
  - Assertions
  - Binary Searches

BACKGROUND

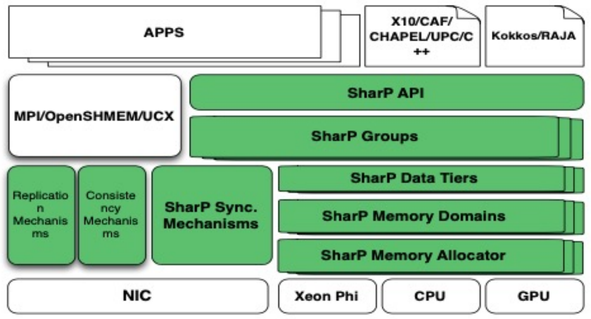
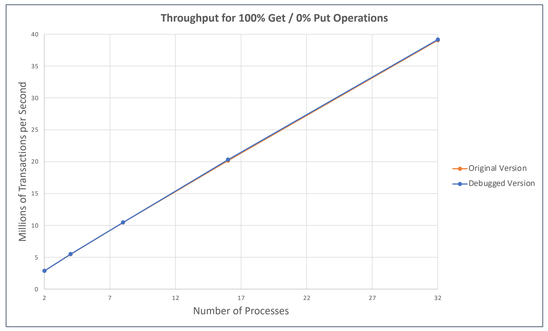


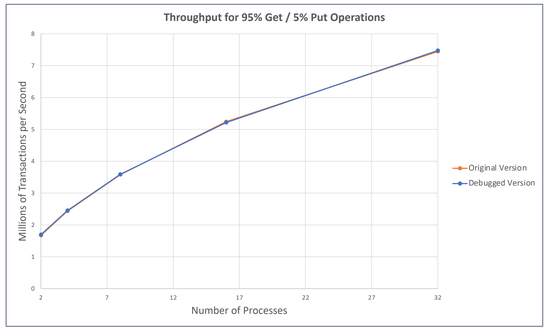
Figure 1 – A graphic representation of the SharP programming model

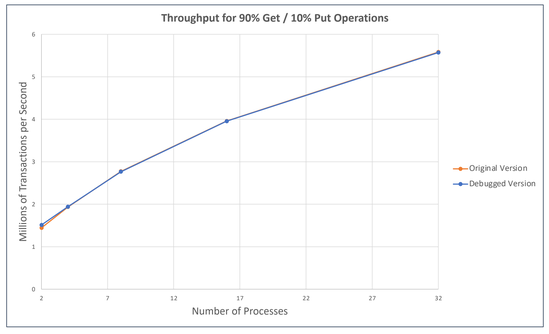
- Data-structure based and data-centric programming construct
  - Global/Local views of data
- Unified and uniform memory abstraction across hierarchical-heterogeneous architectures
  - User-friendly POSIX-like interface
- Library based API and Implementation
  - Supports locality-aware memory management on various memory types including DDRAM, HBM, and Persistent Memory
  - Interoperable with MPI, OpenSHMEM, and OpenMP [1]

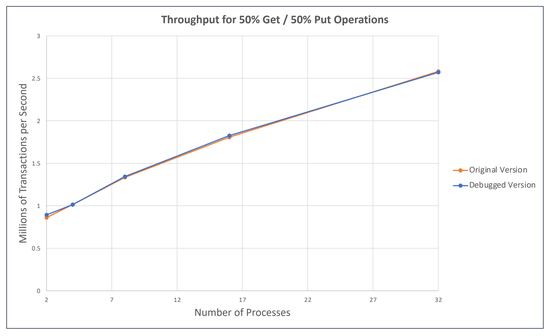
RESULTS

Throughput Benchmark Testing:









CONCLUSION

The first function that required debugging returned the locality information of a processing element (PE) for the distributed hash data structure. It was presenting issues when the data structure wrapped back around but still belonged to a single PE. The issue was traced back to two other functions with logical errors that were utilized by the function of interest. Adding checks in each function was also useful.

The second function that required debugging moves data around in the distributed hash data structure in order to find available space for new insertions. It was presenting intermittent issues when the data spanned more than 3 blocks. The issue was traced back to two other functions with logical errors that utilized the function of interest.

Both issues were successfully addressed and were evaluated against their original versions. We found that our fixes did not add any performance degradation and improved performance in all cases studied.

REFERENCES

1. F. Aderholdt, M. Gorentla Venkata, Z. Parchman. SharP: Towards programming extreme- scale systems with hierarchical heterogeneous memory. *IEEE 46th International Conference on Parallel Processing Workshops*, pp. 145-154, August 2017.

CONTACT

Marie McCord
Email: mam2hu@mtmail.mtsu.edu