Remote Page Faults with a CAPI based FPGA

Joel Nider
IBM Research – Haifa
joeln@il.ibm.com

Yiftach Binyamini
IBM Systems
yiftach@il.ibm.com

Mike Rapoport
IBM Research – Haifa
rapoport@il.ibm.com

CCS Concepts

•**Networks** → **Cloud computing;** Network protocol design; •**Computer systems organization** → *Cloud computing;*

Keywords

CAPI; FPGA; page fault; migration; post-copy

1. PROBLEM

Post-copy VM or container migration requires that the bulk of the memory is transferred after resuming on the destination node [3]. Transferring memory between nodes over a commodity TCP/IP network incurs too much latency, which slows down execution of the application on the destination node.

2. SOLUTION

Live migration can be used in a heterogeneous server environment to take advantage of different attributes of each server to improve energy efficiency or increase performance. Effective post-copy migration depends on a high-speed, low-latency interconnect to move memory pages efficiently between servers [2].

We are building a prototype of a special purpose interconnect that will reduce the latency when transferring memory from a remote machine. We also expect the power consumption of the interconnect to be less than that of a commodity network. The interconnect is designed to be an extension of the system bus, for exclusive use by the operating system for the purposes of VM or container migration. The purpose of the interconnect is to transfer one page of memory from a remote machine to the local machine as quickly as possible, in response to a page fault.

Unfortunately, externally accessible hardware that is directly attached to the system bus does not exist in commodity servers. In order to build a working prototype for testing with real hardware available today, we are using the CAPI protocol as implemented in the IBM POWER8® and OpenPOWER systems. CAPI is intended for providing a cache-coherent view of system memory with attached accelerators over PCIe (I/O bus). That means we can get the required functionality of accessing memory through virtual addresses, despite the latency of the physical layer being less than ideal since it relies on PCIe rather than being directly attached to the system bus.

A CAPI enabled FPGA card includes a unit called PSL (Power Service Layer) which handles all address translation and coherency functions of the CAPI connection. For the initial prototype, the coherent functionality is not used. The PSL unit can be configured to provide only address translation support while sending and receiving traffic as native PCIe DMA packets. Such a configuration reduces the protocol latency and increases bandwidth for applications which do not require coherency support.

For simplicity, the initial prototype involves only two machines connected with FPGA equipped CAPI expansion cards (Nallatech 385A-SoC). The local interface card is connected to the remote interface card over a point-to-point 10Gb/s link using the SerialLite protocol over fiber optics. The card is equipped with 2 such ports, which allows for expansion into a ring topology in the future. All higher level communication (network layer) is through a proprietary protocol consisting of asynchronous requests and responses. Memory read requests are initiated by the local interface card (target) by sending a memory read message to the remote host. To minimize the protocol overhead, only read accesses are to be implemented, which was shown to be effective with RDMA [1].

3. ACKNOWLEDGMENTS

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 688386.

4. REFERENCES

[1] C. Mitchell, Y. Geng, and J. Li. Using one-sided rdma reads to build a fast, cpu-efficient key-value store. In *2013 USENIX Annual Technical Conference*, pages 103–114, San Jose, CA, 2013. USENIX.

[2] J. Nider and M. Rapoport. Cross-isa container migration. In *Proceedings of the 9th ACM International on Systems and Storage Conference*, SYSTOR '16, pages 24:1–24:1, New York, NY, USA, 2016. ACM.

[3] E. Zayas. Attacking the process migration bottleneck. *SIGOPS Oper. Syst. Rev.*, 21(5):13–24, Nov. 1987.

SYSTOR '17 Haifa, Israel

ACM ISBN 978-1-4503-5035-8/17/05…$15.00

DOI: http://dx.doi.org/10.1145/3078468.3078489