**Infiniswap**

**Efficient Memory Disaggregation**

Juncheng Gu, Youngmoon Lee, Yiwen Zhang, Mosharaf Chowdhury, Kang G. Shin

### Background

- Memory-intensive applications are everywhere
  - Low-latency: VoltDB, Memcached
  - Data-intensive: PowerGraph, GraphX

### Underutilization Problem

Google cluster analysis [Charles et al., SoCC 2012]
- 80% memory is allocated, but only 50% is used
- Utilization is imbalanced too

### Memory Disaggregation

Expose cluster memory across server boundaries for performance and efficiency
- In a scalable manner
- With fault-tolerance

### Infiniswap Overview

Scalable and fault-tolerant memory disaggregation using one-sided RDMA
- Without new hardware
- Without modifications to applications and OS
- Page-level operation, slab-level management
- Decentralized design using power-of-many choices

### Slab Placement

Select the least-loaded of the two machines to map slab $S$

### Slab Eviction

Contact up to $E+E'$ machines to evict $E$ slabs

### Disproportionate performance drops with insufficient memory

### Application Performance

- 4X ~ 15.4X higher throughput
- 5.4X ~ 61x lower latency

### Microbenchmarks

- 2-4X higher throughput than Mellanox nbdX
- Zero remote CPU overhead

### Cluster Utilization

- 1.47X higher memory utilization

### Going Forward

Can we improve performance under large failure scenarios?
How to isolate applications when using remote memory?

https://github.com/Infiniswap/infiniswap