DynaHeap: Dynamic Division of DRAM between Heterogeneous Managed Heaps

Iacovos G. Kolokasis, Shoaib Akram, Foivos S. Zakkak, Polyvios Pratikakis, and Angelos Bilas

Big Data Frameworks Need More Memory

- Data grow at an exponential rate, but DRAM scales slower than the data growth
- Existing works extend the managed heap over NVMe SSD, NVM, or remote memory

Static DRAM Division Limitation

- Static DRAM division cannot cope with changing application behavior
- High GC: need space for the first heap
- High IO: need space for cache

DynaHeap: Dynamic Division of DRAM

- No object reference adjustment
- GC scans over the slow tier
- Reduce GC scans over the slow tier
- Need object reference adjustment
- Avoid GC scans over the slow tier
- No object reference adjustment

Controller: Mechanism to adjust memory between H1 and I/O cache

Condition | Action
--- | ---
High GC | Grow H1
High GC and many objects can be moved to H2 | Move H2
High I/O | Shrink H1
Unused memory | -
Next GC event | -
Reset actions | -

Preliminary Results

Key Takeaways

- Applications have different memory requirements at different periods
- Static division of DRAM between H1 and the cache for H2 cannot adapt to dynamic changing application behavior
- DynaHeap is on average 70% better than TeraHeap