Dynamic Block-level Cache Management for Cloud Computing Systems

Dulcardo Arteaga, Douglas Otstott, Dr. Ming Zhao
School of Computing and Information Sciences, Florida International University

Goal: Improve I/O performance of virtual machines (VMs) in cloud systems using caching

Background
- Block-level network storage (iSCSI, NBD, SAN) is commonly used in cloud systems
- Fast VM migrations
- Improved data availability
- Scalability becomes a serious issue as the size of cloud systems increases
- Bottleneck in shared network storage
- Performance interference across VMs

Proposed Solution
- Dynamic block-level client-side caching for cloud computing systems
  - Exploit data locality in VM data access to improve the performance of VMs and the load on shared storage system
  - Utilize the increasing capacity and speed of storage (particularly SSDs) available on the client-side
  - Implement via block-level virtualization to support different cloud storage systems
  - Support flexible, dynamic configuration of cache replacement and write policies

System Design
- Block device virtualization based caching
  - DM-cache: a generic block-level disk cache utility for storage systems
  - Built upon device-mapper, a framework for creating virtual block devices on Linux
  - Can be transparently plugged into an existing IP-SAN/SAN storage system

Shared cache for co-hosted VMs
- Create per-VM virtual caches to differentiate block-level I/Os from different VMs
- Map the different virtual caches to the same physical cache device to maximize cache utilization
- Can support different cache space allocation policies

Proposed Approach
- Cloud system using shared block-level network storage
- Cloud system using dm-cache based client-side block-level caching

Experimental Evaluation
- Eight VM hosts, each with SSD based cache; One shared iSCSI-based network storage server

Conclusion and Future Work
- Conclusions
  - DM-cache effectively uses client-side storage to exploit locality for multiple VMs running on the same physical host
  - SSD-based results show substantial performance improvements for concurrent booting and IOzone runs

Future Work
- Study intelligent algorithms for shared cache space allocation while guaranteeing fairness across all VMs
- Consider the unique characteristics of SSD devices and design optimized cache policies accordingly
- Consider cross-client cooperative caching to further improve caching efficiency and better support VM migration