J. Parallel Distrib. Comput. 74 (2014) 2241–2256

Contents lists available at ScienceDirect

J. Parallel Distrib. Comput.

journal homepage: www.elsevier.com/locate/jpdc

Group-based memory oversubscription for virtualized clouds

Sangwook Kim, Hwanju Kim, Joonwon Lee, Jinkyu Jeong *

College of Information and Communication Engineering, Sungkyunkwan University, South Korea

HIGHLIGHTS

- We identify the limitations of system-wide memory oversubscription in a public cloud.
- We propose a group-based scheme for inter-group isolation and intra-group efficiency.
- We design and evaluate the proposed group-based oversubscription policies.
- We found that the system-wide oversubscription can break inter-group isolation.

ARTICLE INFO

Article history:
Received 28 April 2013
Received in revised form
3 January 2014
Accepted 6 January 2014
Available online 11 January 2014

Keywords:
Virtualization
Memory oversubscription
Resource isolation
Cloud computing

ABSTRACT

As memory resource is a primary inhibitor of oversubscribing data centers in virtualized clouds, efficient memory management has been more appealing to public cloud providers. Although memory oversubscription improves overall memory efficiency, existing schemes lack isolation support, which is crucial for clouds to provide pay-per-use services on multi-tenant resource pools. This paper presents group-based memory oversubscription that confines both mechanism and policy of memory oversubscription into a group of virtual machines. A group is specified as one of service level agreements so that a cloud customer can control the memory management mechanism within its own isolated domain. We introduce group-based memory deduplication and reprovisioning with several policies based on per-group workload behaviors. The proposed scheme is implemented on the KVM-based prototype and evaluated with realistic cloud workloads such as MapReduce and MPI applications. The evaluation results show that our group-based memory oversubscription ensures strict inter-group isolation while achieving intra-group memory efficiency, compared to a system-wide scheme, by adapting oversubscription policies based on per-group workload characteristics.

1. Introduction

Current Infrastructure-as-a-service (IaaS) clouds regard trustworthiness and performance isolation as a major requirement because of their multi-tenant nature. Such isolation demand, however, inhibits the level of oversubscription that allows independent virtual machines (VMs) to share underlying resources, thereby losing the opportunities of gaining more profits. In order to save total cost of ownership, it is important for cloud providers to efficiently share limited resources while guaranteeing service level agreements (SLAs). Several proposals have addressed the issues on isolation over several virtualized resources in cloud environments [26,12,19,11].

Among multi-tenanted resources, memory is a primary inhibitor of oversubscribing data centers due to the nontrivial cost of extension and power consumption [14]. Although modern multi-core processors enable high consolidation ratio, the limitation of memory capacity extension cannot help drawing a line at a lower level of consolidation. In order to alleviate this limitation, many researchers have introduced efficient memory management schemes, such as memory deduplication [36,14,27] and dynamic memory balancing [18,23,45], which allow memory oversubscription by flexibly reprovisioning redundant and unused memory. Memory oversubscription has been considered to be an attractive feature to cloud providers by effectively hosting the increasing number of customers.

Although memory oversubscription improves memory efficiency, it is not trivial to be employed for cloud providers, since existing schemes weaken trustworthiness [27,33] and performance isolation [40,11]. Current memory oversubscription is provided as a system-wide hypervisor (i.e., virtual machine monitor) service, which cannot be controlled and isolated by cloud users. For example, system-wide memory deduplication can impose security breaches by allowing sensitive memory contents to be shared among independent customers [27,33]. In addition, deduplication overhead from one user could interfere the performance of another one who is not being involved in the deduplication. Finally, various

* Correspondence to: Department of Semiconductor Systems Engineering, College of Information and Communication Engineering, Sungkyunkwan University, 85569 Corporate Collaboration Center, 2066, Seobu-ro, Jangan-gu, Suwon, South Korea.

E-mail addresses: swkim@csl.skku.edu (S. Kim), hjukim@csl.skku.edu (H. Kim), joonwon@skku.edu (J. Lee), jinkyu@skku.edu (J. Jeong).

0743-7315/$ – see front matter © 2014 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.jpdc.2014.01.001

policies of memory oversubscription can neither be adapted nor customized to the workload and specific need of each cloud user.

This paper introduces a group-based memory management scheme for virtualized clouds to achieve inter-group isolation as well as intra-group efficiency on memory oversubscription. Fundamentally, the proposed scheme confines the mechanism and policy of memory oversubscription to a group of VMs in order to ensure security and performance isolation between different groups. Group specification is delegated to cloud customers so that they can decide the level of isolation. In order to enable per-group memory management, our scheme isolates memory deduplication and reprovisioning on a group basis. By doing so, memory contents and capacity are securely protected within a group, and oversubscription policy is configurable and adaptive for workload demand of each group.

In order to improve intra-group memory efficiency besides inter-group isolation support, we present three group-based memory oversubscription policies: adaptive scan rate, demand-based memory reprovisioning, and hypervisor-level secondary cache. First, the adaptive scan rate policy dynamically adjusts the rate of memory scanning, which is required to find identical pages for deduplication, for each group by monitoring workloads. Our algorithm takes CPU utilization, swap activities, and deduplication rates into account to figure out effective scan rates. Second, demand-based memory reprovisioning policy distributes per-group surplus memory, which is unused and reclaimed by deduplication, to its group members based on their memory demands. Finally, hypervisor-level secondary cache allows the per-group surplus memory to be used as an exclusive cache that stores pages evicted by VMs within the group.

The group-based oversubscription, however, would degrade system-wide memory efficiency at the expense of strict inter-group isolation. For instance, the group-based scheme prevents a group, which suffers from memory insufficiency, from exploiting underutilized surplus memory of another group. To deal with this limitation, we additionally propose a demand-based overhead migration as a complementary policy for the group-based policies to improve system-wide memory efficiency without breaking inter-group isolation.

The proposed scheme is implemented and evaluated on the KVM hypervisor [22]. We extended KSM (Kernel Samepage Merging) [1] to be a subsystem of Linux *cgroup* [25] for group-based memory deduplication and reprovision, with which the group-based policies are implemented as user-level daemons. The evaluation results show that our group-based memory oversubscription ensures strict inter-group isolation while achieving intra-group memory efficiency over various realistic scenarios using MapReduce, MPI, and a file-intensive workload.

The rest of this paper is organized as follows: Sections 2 and 3 describe the background and motivation behind the proposed scheme, respectively. Sections 4 and 5 explain the design and implementation of the group-based memory management. Section 6 presents evaluation results on our proposed scheme, and the complementary policy for improving system-wide memory efficiency is explained in Section 7. In Section 8, we discuss the applicability of our scheme in cloud environments. Finally, we present related work in Section 9 and conclusions in Section 10.

2. Background

This section presents background on VM memory management and isolation techniques in clouds.

2.1. VM memory management

VM memory management schemes have been mostly focused on memory oversubscription, which allows the hypervisor to allocate more memory to colocated VMs than actual physical memory. The first part of memory oversubscription is to find surplus memory, which is allocated but not actually needed. Memory deduplication and working set estimation can be used in this part. Once surplus memory is secured, it can be reprovisioned to memory-hungry or newly-instantiated VMs in order to increase memory utilization.

Memory deduplication [36,14,27] is a well-known technique that reclaims redundant memory by sharing identical pages. To transparently search identical pages, most schemes periodically scan and compare the contents of existing VM memory. Once identical pages are found, they are merged into a single unique page while redundant ones are reclaimed by the hypervisor. The page table entries referenced to the unique page are marked as copy-on-write, by which modification of each VM is isolated to its private page. During the lifetime between merging and copy-on-write breaking, the hypervisor can maintain additional memory that can be used for other purposes if necessary.

In collaboration with memory deduplication, reclaimed memory can be reprovisioned to existing VMs. The memory ballooning mechanism [36] allows the hypervisor to dynamically increase and decrease VM memory with the aid of guest OS extension. Using this mechanism, surplus memory can be provisioned to a VM that demands more memory beyond its allocation. To identify memory-hungry VMs, existing schemes monitor paging operations such as swap activity [45,40,42]. Reprovisioned memory is used to preserve more working set in memory so that expensive disk I/O operations can be reduced.

2.2. Isolation in clouds

Isolation is an essential support for cloud computing, since cloud providers offer pay-per-use resources to their customers who need certain levels of SLAs. The simplest way of ensuring isolation is to allot dedicated resources to each customer. Many cloud providers, however, strive to save infrastructure costs by oversubscribing their data centers, so that underutilized capacity is effectively provisioned via resource sharing. Therefore, isolation support on shared resources is a key issue cloud providers should deal with in order to maximize their profits with QoS guarantee.

Current IaaS providers have been relying on resource-level isolation policies where SLAs are specified in the form of resource requirement. In this policy, resource capacity for which a customer pays is believed to be isolated by the cloud provider. The resource-level isolation limits the extent to which a cloud provider oversubscribes data center, but ensures strong isolation guarantee. On the other side, several researchers have argued that performance-level isolation can increase oversubscription ratio for more profits [11,26,12]. This policy aims to achieve the performance specified by a customer, regardless of how much resource capacity is provisioned. Although this type of isolation is attractive to providers, it is challenging to estimate dynamically varying performance and identify relationship between performance and various types of resources.

3. Motivation

In this section, we first show the significance of the overhead arising from memory deduplication in terms of performance interference. Next, we estimate the impact of group-based memory deduplication in aspects of shareable memory since serious reduction of shareable memory can offset the advantage of memory isolation. Then, we clarify the need for group-based customization of memory oversubscription by describing the workload heterogeneity with inherent memory characteristics in virtualized clouds. Finally, we argue the limitation of existing memory oversubscription schemes from the perspective of isolation and flexibility.