Autonomic Computing Architecture for Real-Time Medical Application Running on Virtual Private Cloud Infrastructures

Yong woon Ahn, Albert Mo Kim Cheng
Department of Computer Science
University of Houston
4800 Calhoun Road, Houston, Texas, U.S.A.
+1-713-743-3350
{yahn, cheng}@cs.uh.edu

ABSTRACT

Cloud computing with virtualization technologies has become a huge trend which attracts academia and information technology industries because of its cost-efficiency. It has changed paradigms of development, release, and maintenance of diverse types of software and service. However, this big movement has not been applied to real-time applications yet because deploying real-time applications on the cloud infrastructures arouses many controversies because of numerous uncertainties from sharing physical resources. It is not trivial to apply conventional scheduling techniques for real-time systems to cloud infrastructures without further considerations. All virtual machines (VMs) must be controlled by the Virtual Machine Monitor (VMM) which is centralized and has a primary role to share physical resources fairly with all VMs in the same physical machine (PM). For best-effort applications, this fair resource sharing policy works well and end-to-end Quality of Service (QoS) is promised by the service level agreement (SLA) with reasonable delay windows. However, for real-time applications with aperiodic hard- and soft-real-time tasks, this mechanism has serious weaknesses. Although VMMs of most cloud infrastructures have auto-scaling and load-balancing mechanisms, these are unpredictably slow to accept urgent aperiodic-real-time tasks because of its fair resource sharing policy. Therefore, it is necessary to propose another approach to satisfy deadline constraints of real-time tasks transferred from remote locations. In this research, we focus on cloud medical applications processing sensitive patient data with deadline constraints. We propose feasible solutions to reserve computing and networking resources with an autonomic computing architecture in the virtual private cloud infrastructures.

Keywords

Auto-Scaling, Cloud Computing, Real-Time Systems, Medical Application, Virtualization, Autonomic-Computing

1. INTRODUCTION

When you run your medical application handling sensitive data on the public cloud infrastructure, your primary concern is how to protect your data from other computing components running with your instances because they have to share limited physical computing power, storage, and network bandwidth with their unknown neighbors. This problem can be resolved by using the virtual private cloud infrastructure [1] which has specialized mechanisms to support virtual private network protecting sensitive data. However, the virtual private network cannot resolve issues to support real-time applications with periodic and aperiodic tasks streamed from remote locations because a centralized VMM still needs to control VMs with its auto-scaling and load-balancing mechanisms satisfying its fair resource sharing policy. Therefore, a new approach is required to support reserving virtual resources on time.

*Supported in part by the National Science Foundation under Awards No. 0720856 and No. 1219082.

2. AUTONOMIC COMPUTING

In order to design a cloud infrastructure to run real-time applications on it, the most important consideration is how to scale up virtual resources to process real-time tasks, and scale down them to save costs. With conventional VMMs, it would be hard to monitor and control VMs which possibly are located on different physical machines. Therefore, we consider a distributed VM monitoring method with an autonomic computing architecture [2] for virtual private cloud infrastructures. An autonomic computing architecture has four phases to process tasks input from managed resources as shown in Figure 1. Also it can adjust its own system parameters to optimize processing performance and internal resource usages. We focus on the fact that these managed resources can be other autonomic computing architectures to build proper orchestration. In our research they can be multiple VMs. Moreover, we design each VM to be greedy and accept more tasks to process, and it always tries to steal real-time tasks from its child VMs, if its parent VM can process the same type of task. As a result of our approach, each VM and eventually each group of VMs can be optimized to process real-time tasks with lower costs. An auto-scaling mechanism also can be implemented with this approach. If one VM cannot handle tasks from its managed resources, it can invoke a new child VM. Since, the parent VM is greedy, the child VM is terminated when it has few jobs.

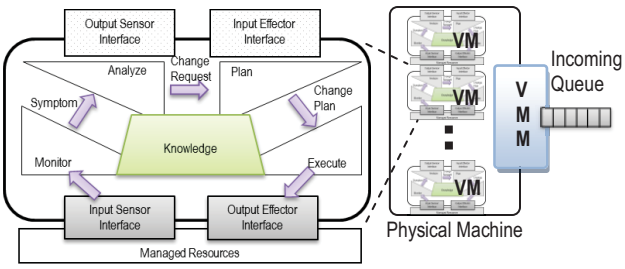

Figure 1. Autonomic computing architecture for VMs

3. CONCLUSION

In this research, we focus on running real-time medical applications on the private cloud infrastructures. In order to design and implement a scalable and reliable auto-scaling mechanism, we apply an autonomic computing concept to our system.

4. REFERENCES

[1] Miyamoto, T., Hayashi, M., Nishimura, K. Sustainable Network Resource Management System for Virtual Private Clouds. IEEE CloudCom, pp. 512-520. 2010.

[2] IBM, "Autonomic computing: IBM's perspective on the state of information technology," IBM Corporation, 2001.