Integrating AGILE with CloudScale

Bharath Venkatesh
bharat@ncsu.edu

Devendra D. Chavan
ddchavan@ncsu.edu

Girish Prabhu H.
gprabhu@ncsu.edu

Department of Computer Science
North Carolina State University

ABSTRACT

Elasticity is a key feature in cloud computing system as it allows dynamic resource scaling in order to keep application performing at the required level. This paper presents an application agnostic approach for elastic scaling of resources with minimum provisioning costs, such that application service level objective (SLO) violations are minimized. Our approach integrates earlier work done on localized resource scaling in CloudScale with the distributed resource scaling in AGILE. This combined approach allows the system to make both short term and medium term resource scaling predictions to effectively minimize application SLO violations and improve system performance. We have implemented the integrated system on the KVM virtualization platform. Tests using RUBiS and Cassandra show that the integrated system achieves higher SLO conformance than either system alone.

1. INTRODUCTION

IaaS providers often opt for overbooking [15] techniques to achieve better resource utilization and provide a resource capping to achieve performance isolation. However if they provide a very high resource cap, it leads to excessive resource wastage and the provider has to pay for these wasted resources. Conversely, if the resource cap is too low, the applications suffer from performance degradation which leads to SLO violations. These violations are very expensive to the providers due to the penalties it imposes. To overcome these problems, we need an elastic approach to adjust the resource cap dynamically based on requirements.

There is a difference in perception of elasticity between users and providers. For a user, cloud is a remote location that offers infinite resources which can be accessed via set of exposed APIs [6]. Providers are responsible for acquiring, managing and updating the cloud infrastructure. They must also provide some mechanism to enable elasticity. Some providers leave it upto the user to modify resource requirements manually. Meanwhile, other providers incorporate automated services that monitor the current load and allocate resources accordingly. There are two ways to achieve automation. *1*) Reactive approach: it is based on *Rule-Condition-Action* approach wherein specific action is triggered when certain rules are satisfied. This is a popular approach followed by leading providers such as Amazon [1], Rightscale [3] and in several academic works [10, 12]. The drawback with this approach is that the system behaviour is predefined in nature. *2*) Predictive approach: this uses mathematical/analytical models and heuristics to predict system behavior and proactively decide how to scale the resources [14, 8, 10].

CloudScale [13] provides a robust prediction driven approach for scaling resources based on the need of applications within in the single host. It uses a hybrid approach developed in PRESS [7] to generate predictions for both periodic and non periodic workload patterns. The system is implemented on top of the Xen [5] virtualization platform. Predictive migration is employed when all the available resources are insufficient to meet the scaling requirement. The lead time to start up a new application server instance is usually large and this leads to large latencies in distributed resource scaling. Thus arises the need for medium term prediction techniques to handle scaling in a distributed environment. AGILE [11] provides a medium term resource demand prediction which can provide enough time to scale up the server pool before the application's SLO gets affected. It uses dynamic VM cloning that triggers the live cloning before the application enters the overload state. The pre-copy live cloning process itself if augmented with rate control over data transfer to minimize interference with existing VMs.

CloudScale provides an efficient way of scaling resources within a host, the same approach proves insufficient in a distributed environment due to the higher latencies. AGILE effectively complements the short term localized resource scaling by replicating VMs to scale the performance. In our work, we propose to leverage on the benefits of both the systems to build a system that is able to scale the resources both on the host as well as in the distributed environment.

The objective of our work are as follows:

- Implementing CloudScale in KVM [9] to achieve predictive resource scaling on a single host.
- Integrate CloudScale with the distributed resource scaling provided by the AGILE system

- Evaluate the performance of the combined system against the existing individual systems using RUBiS [4] multi-tier online auction benchmark [4] and the Cassandra [2] key-value store system,

2. PROPOSED APPROACH

CloudScale effectively relieves local (per server) application loads and extends the online resource demand prediction model developed in PRESS that uses *Fast Fourier transforms (FFT)* to determine cyclic workload patterns in resource usage and predicts future resource requirements. For acyclic workload patterns, it applies *Markov Chain model.* While AGILE focuses on resource scaling in distributed environments (scaling the server pool) with high instantiation latencies for VMs. It uses wavelet based approach to achieve medium term predictions. Our work is focussed on integrating the local elastic resource scaling offered by CloudScale with the distributed resource scaling in AGILE. CloudScale is implemented on the Xen virtualization platform. An initial part of our work would also be to reimplement it on KVM.

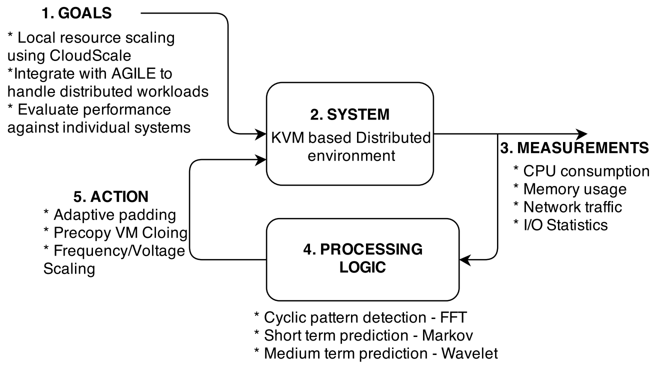
1. GOALS
* Local resource scaling using CloudScale
*Integrate with AGILE to handle distributed workloads
* Evaluate performance against individual systems
2. SYSTEM
KVM based Distributed environment
3. MEASUREMENTS
* CPU consumption
* Memory usage
* Network traffic
* I/O Statistics
4. PROCESSING LOGIC
* Cyclic pattern detection - FFT
* Short term prediction - Markov
* Medium term prediction - Wavelet
5. ACTION
* Adaptive padding
* Precopy VM Cloing
* Frequency/Voltage Scaling

Figure 1: Feedback loop for the proposed approach

Figure 1 shows the overall structure of the proposed approach. The model periodically samples system performance metrics such as CPU consumption, memory usage, network traffic and disk I/O statistics. Based on the inputs obtained from these measurements, it performs the three predictions predictions for short term and medium term resource scaling. In addition, it also applies adaptive padding to compensate for underestimation errors arising due to bursty traffic. Based on the needs predicted by the processing logic module, the model either scales the individual VM resources or employs live pre-copy VM cloning technique to replicate the source VM.

2.1 Delineation of duties

Initial collective tasks

- Understanding KVM and Xen hypervisors
- Understanding the prediction models: Fast Fourier transforms, Markov chain and wavelet based prediction
- System design

Individual tasks

Component Name	Developer	Tester
Prediction error correction	bharat	ddchavan
Reactive error correction	ddchavan	gprabhu
Scaling conflict prediction	gprabhu	bharat
Local conflict handling	bharat	gprabhu
Cap adjustment	gprabhu	ddchavan
Integration	bharat	
Evaluation - Cassandra	ddchavan	
Evaluation - RUBiS	gprabhu	
Final Report	ALL	

Table 1: Delineation of duties - Individual tasks

3. REFERENCES

[1] "Amazon Web Services". http://aws.amazon.com. Accessed: 2014-02-02.

[2] "Apache Cassandra". http://cassandra.apache.org/. Accessed: 2014-02-02.

[3] "RightScale". http://www.rightscale.com. Accessed: 2014-02-02.

[4] "RUBiS Online Auction system". http://rubis.ow2.org/. Accessed: 2014-02-02.

[5] P. Barham, B. Dragovic, K. Fraser, S. Hand, T. Harris, A. Ho, R. Neugebauer, I. Pratt, and A. Warfield. Xen and the art of virtualization. *ACM SIGOPS Operating Systems Review*, 37(5):164–177, 2003.

[6] G. Galante and L. C. E. d. Bona. A survey on cloud computing elasticity. In *Utility and Cloud Computing (UCC), 2012 IEEE Fifth International Conference on*, pages 263–270. IEEE, 2012.

[7] Z. Gong, X. Gu, and J. Wilkes. Press: Predictive elastic resource scaling for cloud systems. In *Network and Service Management (CNSM), 2010 International Conference on*, pages 9–16. IEEE, 2010.

[8] N. Huber, F. Brosig, and S. Kounev. Model-based self-adaptive resource allocation in virtualized environments. In *Proceedings of the 6th International Symposium on Software Engineering for Adaptive and Self-Managing Systems*, pages 90–99. ACM, 2011.

[9] A. Kivity, Y. Kamay, D. Laor, U. Lublin, and A. Liguori. kvm: the linux virtual machine monitor. In *Proceedings of the Linux Symposium*, volume 1, pages 225–230, 2007.

[10] H. C. Lim, S. Babu, and J. S. Chase. outomated control for elastic storage. In *Proceedings of the 7th international conference on Autonomic computing*, pages 1–10. ACM, 2010.

[11] H. Nguyen, Z. Shen, X. Gu, S. Subbiah, and J. Wilkes. Agile: elastic distributed resource scaling for infrastructure-as-a-service. *Detail*, 3:0.

[12] S. Parekh, N. Gandhi, J. Hellerstein, D. Tilbury, T. Jayram, and J. Bigus. Using control theory to achieve service level objectives in performance management. *Real-Time Systems*, 23(1-2):127–141, 2002.

[13] Z. Shen, S. Subbiah, X. Gu, and J. Wilkes. Cloudscale: elastic resource scaling for multi-tenant cloud systems. In *Proceedings of the 2nd ACM Symposium on Cloud Computing*, page 5. ACM, 2011.

[14] B. Trushkowsky, P. Bodík, A. Fox, M. J. Franklin,

M. I. Jordan, and D. A. Patterson. The scads director: Scaling a distributed storage system under stringent performance requirements. In *FAST*, pages 163–176, 2011.

[15] B. Urgaonkar, P. Shenoy, and T. Roscoe. Resource overbooking and application profiling in shared hosting platforms. *ACM SIGOPS Operating Systems Review*, 36(SI):239–254, 2002.