Marrying HPC and Cloud for Long Term Happiness

Apoorve Mohan[†], Ravi S. Gudimetla[†], Ata Turk[*], Sourabh Bollapragada[†], Rajul Kumar[†], Jason Hennessey[*], Evan Weinberg[*], Dimitri Makrigiorgos[*], Christopher N. Hill[‡], Gene Cooperman[†], Peter Desnoyers[†], Richard Brower[*] and Orran Krieger[*],
[*]Boston University, Boston, MA
[†]Northeastern University, Boston, MA
[‡]Massachusetts Institute of Technology, Cambridge, MA

***Abstract*—Traditional high performance computing (HPC) clusters, with deep job submission queues, by construction are always almost fully utilized. On the other hand, the cloud has time-varying workloads, and the cloud business model depends on being underutilized to instantly support all customer requests. A marriage of these two environments could provide additional resources to HPC users while offering increased utilization to the cloud. In this poster we present the frameworks we built to enable a successful symbiotic co-existence of HPC and the cloud and showcase the benefits achievable with a prototype deployment.**

I. Introduction and Motivation

Cloud providers plan their hardware purchases by considering peak user utilization and adding some slack so as to not turn down customers and to be able to guarantee as much isolation as possible. This leads to under-utilized cloud deployments and an increase in total cost of ownership for cloud systems. Even though solutions such as auctioning unused cpu cycles (e.g., Amazon spot market [1]) or offering short-lived preemptible virtual machines (e.g., Google preemptible instances [2]) can mitigate the impact of this under-utilization problem, they do not completely address it. Studies show that even large public cloud datacenters have utilization levels below 50% [3].

High Performance Computing (HPC) systems, on the other hand, have a totally different and somewhat complementary workload pattern. HPC workloads generally use a batch submission model, with the inherent understanding that the resources they demand may not be assigned to them right away. Thus, HPC datacenters generally boast very high utilization levels ($\sim$90%) [] but can suffer from long queue wait times [4].

Current HPC deployments consist of a number of public-funded large-scale shared HPC infrastructures that have over-subscribed resources and a long-tail of small-scale dedicated deployments that do not benefit from the economies of scale and shared resources. The on-demand access model of cloud computing is appealing espacially to this second category of workloads.

In this poster we present our efforts towards designing a combined HPC and cloud deployment system that has significantly improved overall utilization and provides sustainable, scalable, cheap and performant bare-metal resources to HPC. The solutions we offer reside in the hardware-layer, and provide fast exchange of servers between HPC and cloud deployments. We designed two hardware-layer services, namely the Hardware Isolation Layer (HIL) [5] and the Bare-Metal Imaging Service [6]. In this poster we showcase with a proof of concept prototype that these services combined with a real-time monitoring system can support elastic cloud and HPC deployments and offer enormous benefits to both environments.

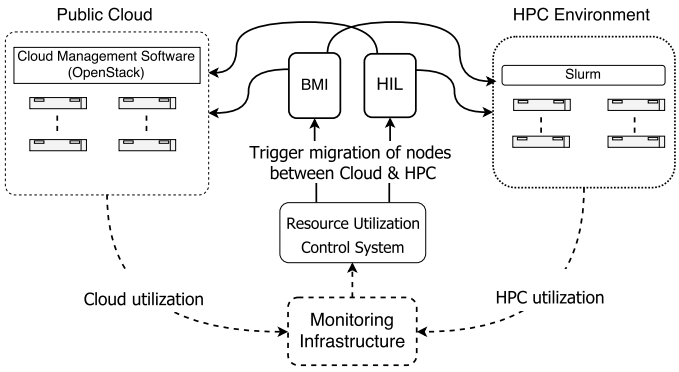

Fig. 1. Infrastructure of the proof of concept deployment.

II. Architecture and Proof of Concept Deployment

Figure 1 displays the prototype we developed to evaluate our approach. Our monitoring service constantly monitors the overall utilization levels of the cloud, the HPC deployment and the datacenter. The Resource Utilization Control System (RUCKS) retrieves the overall utilization of the datacenter and the cloud deployment from the monitoring service, and uses the bare-metal imaging (BMI) and Hardware Isolation Layer (HIL) services to migrate nodes from one cluster to the other based on resource utilization thresholds. If the cloud cluster's utilization level reduces below a certain threshold RUCS dynamically migrate the least utilized physical node(s) from the cloud cluster to the HPC cluster using HIL and BMI services while making sure that the VM's that were running on the least utilized physical nodes gets live-migrated to other nodes present in the cloud cluster. Our assumption in here is that HPC job queues are long enough to maintain the utilization level of the HPC cluster at a high level thus this migration eventually will increase the utilization of the entire data center. If the monitoring service observes an increase in the cloud cluster utilization (beyond a specified threshold),

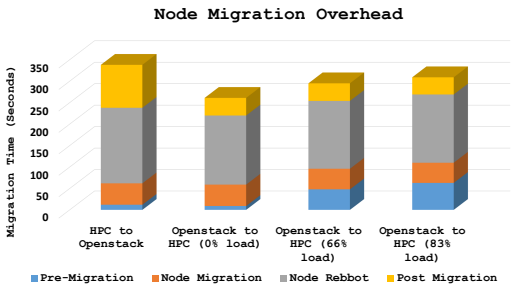

Fig. 2. Node migrating times (seconds).

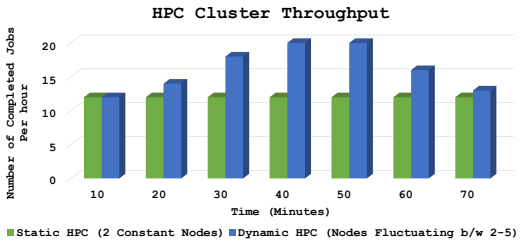

Fig. 3. HPC throughput comparison of static HPC-Cloud clusters and dynamic HPC-Cloud with migration.

RUCS dynamically move nodes from HPC cluster back to the cloud cluster.

III. Experiments and Results

In our experiments and analyses we used OpenStack as our cloud management system and we used Slurm as our HPC management system. In Figure 2, we present the operation times in seconds while we are migrating nodes from an HPC deployment to a cloud deployment (HPC to OpenStack) and from a cloud deployment to an HPC deployment (OpenStack to HPC). For all migration scenarios, the amount of time spent for node migration (operations associated with HIL) and node rebooting is the same. For the case of HPC to Cloud migration, the amount of time spent for making the node to be migrated ready is relatively low even though we assume that the HPC nodes are always running with high (above 90%) utilization. However, the amount of time spent for integrating a new node into the cloud management system is significantly high. All in all, migrating a node from HPC to cloud takes around five and a half minute. On the other hand, for the case of OpenStack to HPC migrations, we considered multiple cases where the node to be migrated had different loads.

The total number of nodes in the cluster used for the following experiments is 11. In the static deployment of cloud and HPC, the HPC cluster has three nodes (including the Slurm controller node) and the OpenStack cluster has eight nodes one of which is the OpenStack controller. We approximate a time-varying workload for the cloud by using a cosine pattern of demand on the OpenStack cluster. To

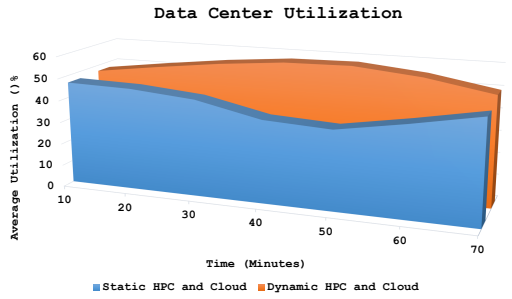

Fig. 4. Utilization comparison of static HPC-Cloud clusters and dynamic HPC-Cloud with migration.

approximate the cosine curve, we sampled it at ten minute intervals and stopped or started some of the VMs running on the cloud based on the sampled value. Due to the variance generated by the time-varying workload on the OpenStack, the number of worker nodes on the HPC cluster in the dynamic deployment varies between two and five.

In Figure 3, we compare the throughput of a static HPC deployment with a dynamic HPC system that shares nodes with an OpenStack deployment. The number of nodes on the static deployment is fixed to two, whereas, based on the load on the OpenStack deployment the number of nodes in the dynamic deployment vary. The observed throughput values are in line with the amount of load OpenStack cloud observes.

Finally in Figure 4 we present the overall utilization of the whole cluster supporting both the HPC and Cloud systems. As seen from the figure, the overall utilization of the dynamic deployment is comparably higher compared to static deployment.

IV. Conclusion

In this study we presented our initial steps for building a framework that provides additional resources to HPC users while offering increased utilization to the cloud. Our initial findings indicate the usefulness of this approach.

References

[1] "Amazon ec2 spot instances," 2016. [Online]. Available: https://aws.amazon.com/ec2/spot/

[2] "Google preemptible vm instances," 2016. [Online]. Available: https://cloud.google.com/compute/docs/instances/preemptible

[3] NRDC and Anthesis, "Data Center Efficiency Assessment: Scaling up energy efficiency across the Data Center Industry: evaluating Key Drivers and Barriers," NRDC and Anthesis, Tech. Rep., 2014.

[4] A. Marathe, R. Harris, D. K. Lowenthal, B. R. de Supinski, B. Rountree, M. Schulz, and X. Yuan, "A comparative study of high-performance computing on the cloud," in *Proceedings of the 22Nd International Symposium on High-performance Parallel and Distributed Computing*, ser. HPDC '13. New York, NY, USA: ACM, 2013, pp. 239–250. [Online]. Available: http://doi.acm.org/10.1145/2462902.2462919

[5] J. Hennessey, S. Tikale, A. Turk, U. Kaynar, C. Hill, P. Desnoyers, and O. Krieger, "Hil: Designing an exokernel for the data center," 2016, submitted to ACM Symposium on Cloud Computing (SoCC 16).

[6] A. Turk, R. S. Gudimetla, E. U. Kaynar, J. Hennessey, S. Tikale, P. Desnoyers, and O. Krieger, "An experiment on bare-metal bigdata provisioning," in *8th USENIX Workshop on Hot Topics in Cloud Computing (HotCloud 16)*, 2016.