2012 IEEE Fifth International Conference on Cloud Computing

A Performance Study on the VM Startup Time in the Cloud

Ming Mao
Department of Computer Science
University of Virginia
Charlottesville, VA 22904
ming@cs.virginia.edu

Marty Humphrey
Department of Computer Science
University of Virginia
Charlottesville, VA 22904
humphrey@cs.virginia.edu

Abstract—**One of many advantages of the cloud is the elasticity, the ability to dynamically acquire or release computing resources in response to demand. However, this elasticity is only meaningful to the cloud users when the acquired Virtual Machines (VMs) can be provisioned in time and be ready to use within the user expectation. The long unexpected VM startup time could result in resource under-provisioning, which will inevitably hurt the application performance. A better understanding of the VM startup time is therefore needed to help cloud users to plan ahead and make in-time resource provisioning decisions. In this paper, we study the startup time of cloud VMs across three real-world cloud providers – Amazon EC2, Windows Azure and Rackspace. We analyze the relationship between the VM startup time and different factors, such as time of the day, OS image size, instance type, data center location and the number of instances acquired at the same time. We also study the VM startup time of spot instances in EC2, which show a longer waiting time and greater variance compared to on-demand instances.**

Keywords-cloud computing; VM startup/acquisition/spinup time; performance study; spot instances;

I. INTRODUCTION

One great advantage of the cloud is the elasticity, the ability to dynamically provision resources in response to demand. Based on the workload information and performance requirements, the users can acquire more cloud VMs to handle workload surges, or release cloud VMs to avoid resource over-provisioning. This on-demand scalability feature has attracted many people to join the cloud community to help them handle the dynamical workload and reduce their cost. However, one important fact in this dynamic process is that though cloud users can make their acquisition requests at any time, it may take some time for the acquired VMs to be ready to use. Cloud providers need time to find a spot to provision the VM in their data centers, to allocate resources (e.g. IP addresses) to the VM and to copy/boot/configure the OS image. We call this time period the *VM startup time*. Cloud users have been aware of this delayed resource availability issue and complained about the unexpected long waiting time [1][2]. It slows down their development progress, and more importantly, hurts the performance of their cloud applications. The advantages of the cloud are greatly discounted in such cases. The research community has started to work on this problem and proposed several techniques to speed up the VM provisioning process [3][4][5][6]. In this paper, we perform a systematic study on the cloud VM startup time across three cloud providers – Amazon EC2, Windows Azure and Rackspace (or simply EC2, Azure and Rackspace). We report the numbers and facts, analyze the relationship between the VM startup time and different factors, compare the three cloud providers and make recommendations whenever possible.

This performance study is important. Understanding the VM startup time will help cloud users to plan ahead and make better resource provisioning decisions. This information is particularly important to the time-critical applications that rely on the cloud's on-demand resources and is crucial to the cloud auto-scaling mechanisms to make correct decisions [7][8]. As pointed out by the work "Time is money, the value of on-demand [9]", understanding the VM startup time also helps to determine the requirements of the resource usage forecasts. If the demand cannot be forecasted accurately further out than the VM startup time, the value of on-demand diminishes. Currently, we do not see any complete study on the VM startup time in the cloud. [10] covers five instance types for single- and multiple-instance requests in EC2. [11] covers the WebRole and WorkerRole when Windows Azure is in the community technology preview phase. In both works, they do not consider factors like time of the day, OS image size, instance type and data center location, which will affect the VM startup time as well. They do not cover the newly published services like spot instances and VMRole. Since these studies are performed two years ago, it is also interesting to see whether there is any improvement on the VM startup time in the past two years. In this paper, the measurement of the VM startup time is not relied on the status tags provided by the cloud providers, which are sometimes confusing and imprecise. For example, a "running" instance does not necessarily mean the acquired VM is ready to use. It could be still in the OS downloading phase. Instead, we use the duration from the time of issuing VM acquisition requests to the time that the acquired instances can be logged in remotely as the VM startup time. This is a more precise and direct measurement than the status tags.

In addition to reporting the numbers and facts collected in the experiments, we try to analyze the data, extract useful information and compare the cloud providers whenever possible. Here are the highlights of our findings.

978-0-7695-4755-8/12 $26.00 © 2012 IEEE
DOI 10.1109/CLOUD.2012.103

- Within each cloud provider, the VM startup time of both Linux and Windows machines are independent of time of the day.
- In EC2 and Rackspace, Windows instances take around 9 times longer than Linux instances. In Azure, all three Role instances show similar performances.
- The size of the OS image can largely affect the VM startup time. For all three cloud providers, the VM startup time increases linearly as the image size increases.
- The VM startup time is also affected by the instance type. In Azure and Rackspace, the VM startup time goes longer as the instance type goes larger. However, this is not the case in EC2.
- Within each cloud provider, the VM startup time does not show significant differences across different data center locations. One exception is that in EC2, the newly established data center shows a slower VM startup time (20%) and greater variance than the existing data centers.
- In EC2, the VM startup time is relatively constant across all instances when requesting a pool of VMs to start. However, in our experiments, we frequently observed that this was not the case in Windows Azure, as the last VM instance to come on-line sometimes took significantly longer than the first instance.
- For all the factors, spot instances show a longer VM startup time and greater variance compared to the on-demand instances. There's a significantly longer waiting time for the cloud provider to serve a spot instance request, while the actual VM provisioning and booting time are consistent with on-demand instances. Moreover, we do not see a significant correlation between the VM startup time and the real-time spot price.
- Instance acquisition requests are not always successfully served by the cloud providers. In our experiments, Rackspace had a higher failure rate (8%) than EC2 (0.8%) and Azure (0.4%).
- The VM release time is not affected by the OS image size, instance type or data center location.
- Compared to the related studies conducted two years ago, Azure shows a 200-second improvement and smaller variance on the VM startup time, while EC2's performance does not change.

The rest of the paper is organized as follows. Section II describes the three cloud providers we have chosen and the experiment setup process. Section III details and analyzes the experiment results based on different factors, such as time, data center location, instance type, etc. Section IV discusses the related work. Finally, we conclude the paper and discuss the future work in Section V.

II. CLOUD PROVIDERS & EXPERIMENT SETUP

In this performance study, we choose three cloud providers as our experiment subjects. They are Amazon EC2[12], Windows Azure[13], and Rackspace[14]. We choose these three cloud providers because they are popular and have been continuously ranked as the top 10 cloud providers [15][16]. EC2 and Rackspace are well-known Infrastructure-as-a-Service (IaaS) cloud providers. They can provision VMs with different hardware configurations based on the user requests. Azure is a well-known Platform-as-a-Service (PasS) cloud provider. Initially, Azure offers WebRole and WorkerRole for hosting front-end web applications and processing backend tasks. Recently it allows users to deploy a Windows image prepared offline called VMRole in the cloud, in which cloud users can control the whole software stack and log in remotely. This is essentially an IaaS type of service. And not like other PasS cloud providers, even for their WebRole and WorkerRole services, Azure has offered APIs and enabled remote desktop connections to talk to the hosting operating system, which functions more like a VM. For all the roles, users can choose appropriate instance types and monitor the life-cycles of the acquired instances. Because of these practical considerations, we include Azure in this study.

In this study, we will examine the VM startup time based on different factors. These factors include time of the day, OS image size, instance type, data center location and the number of the instances requested at the same time. In each experiment, we use the following default parameters in TABLE I. Currently, EC2 offers twelve instance types, Azure offers five instance types and Rackspace offers seven instance types. The default instance types are m1.small, Small and Type IV for each cloud provider, these three instance types have similar but not exactly the same computing power. OS images include both Linux and Windows for EC2 and Rackspace. For Azure, the default WebRole and WorkerRole applications are the template applications included in the Azure SDK. The default OS image for VMRole is plain Windows Server 2008R2 without any user application installed. For high availability and small communication latency, cloud providers currently have globally distributed data centers to host user VM instances. By default, our experiments are conducted in the availability zone 1 of the US East region for EC2 and South Central US for Azure. Rackspace currently does not support the VM location options.

TABLE I. DEFAULT VM TYPE, OS & LOCATION

Type	OS Image	Location
Amazon EC2		
m1.small	Linux(Fedora) ami-48aa4921	us-east-1a
m1.small	Windows (Win Server 2008) ami-fbf93092	us-east-1a
Windows Azure		
Small	WebRole default WebRole app in Azure SDK	South Central US
Small	WorkerRole default WorkerRole app in Azure SDK	South Central US
Small	VMRole Win Server 2008R2	South Central US
Rackspace		
Type IV	Linux (Fedora) flavor 71	N/A
Type IV	Windows (Win Server 2008R2) flavor 28	N/A

To describe the life-cycles of the cloud VM instances, cloud providers use a set of status tags to indicate the states of the acquired VM instances. The state transition of a VM instance from acquisition to release is summarized in Figure

1. Clearly, cloud providers define different phases/states for their VM instances. Moreover, the meanings of the states are sometimes confusing. For example, a “running” instance does not necessarily mean that the instance is ready to use. The instance could still be downloading the OS image or in the booting process. To make the definition of the startup time consistent across the three cloud providers and provide a more precise measurement, in our experiment, we ignore the status tags. Instead, we use the duration from the time of issuing VM acquisition requests to the time that the acquired instances can be logged in remotely as the VM startup time. For Linux, we use the first successful ssh login time. For Windows, we use the first successful remote-desktop connection time. For Azure WebRole and WorkerRole, we use the first successful http request time served by the WebRole instances and the first logging time by the WorkerRole instances as the VM startup time. The detailed definition can be found in TABLE II.

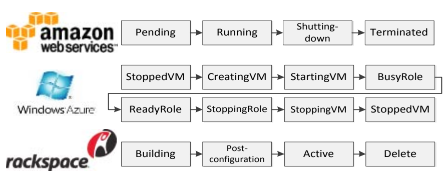

Figure 1. The state transition of Cloud VMs.

TABLE II. DEFINITION OF VM STARTUP TIME

VM	Definition of VM startup time
Linux VMs	The first successful ssh login
Windows VMs	The first successful remote desktop connection
WebRoles	The first successful http request
WorkerRoles	The first successful logging

This performance study is conducted from Oct 15th, 2011 to Feb 15th, 2012, totally a four-month period. We write a client to collect VM startup information from the three cloud providers. The client runs on one machine of the eScience group at CS department, UVa and periodically sends POST requests to create VM instances using the cloud provider API.

III. EXPERIMENT RESULTS AND ANALYSIS

A. By Time

Our first experiment is to measure the relationship between the cloud VM startup time and time of the day. We show both the quantiles and the average of the VM startup time for Linux and Windows machines in the three figures below. For all three cloud providers, the VM startup time is independent of time of the day. We do that see any clear patterns. This may imply that the VM startup time is unlikely affected by the business open or close hours. For both EC2 and Rackspace, it is around 9 times faster to acquire Linux machines than Windows machines. The main reason is that Windows image is much larger than the Linux image and the data transfer time dominates the VM startup time. The effects of OS image size on the VM startup time will be explained in more details in the next experiment. For Azure, we see that WebRole, WorkerRole and VMRole instances show very similar startup performance. Therefore, we show both the quantiles and the average for WebRole instances, but only the average for WorkerRole and VMRole instances. On average, the VM startup time of WebRole instances is 20 seconds shorter than the VMRole instances, but 30 seconds longer than the WorkerRole instances. VMRole takes the shortest time to startup since it is a plain Windows image without any other software/service installed (e.g. no IIS as in WebRole). The results also show that Azure has a 200 second improvement and smaller variance on the VM startup time than two years ago [11].

TABLE III. AVERAGE VM STARTUP TIME

Cloud	OS	Average VM startup time
EC2	Linux	96.9 seconds
EC2	Windows	810.2 seconds
Azure	WebRole	374.8 seconds
Azure	WorkerRole	406.2 seconds
Azure	VMRole	356.6 seconds
Rackspace	Linux	44.2 seconds
Rackspace	Windows	429.2 seconds

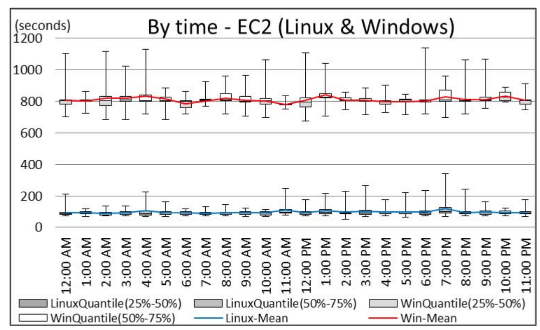

Figure 2. EC2 VM startup time – by time of the day.

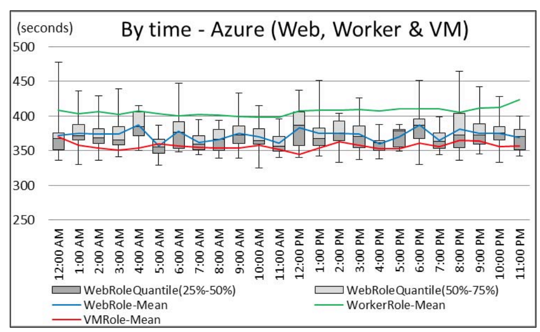

Figure 3. Azure VM startup time – by time of the day.

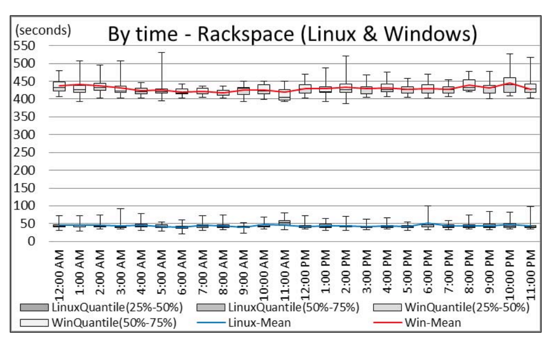

Figure 4. Rackspace VM startup time – by time of the day.

B. *By OS Image Size*

In this experiment, we measure the VM startup time by different OS image sizes in the cloud. For EC2 and Rackspace, we use the default Linux OS image as the base image, add more files under the root directory and then save it as a larger OS image. The four images are 512M, 1G, 2G and 4G by size. For Azure WebRole, we add more files to the default WebRole application. Because the maximum size of an Azure application package is 600M, the four applications used in this experiment are 4M, 32M, 256M and 512M. The VM startup time is shown in Figure 5. Figure 6. and Figure 7. For all three cloud providers, the VM startup time increases linearly as the image size increases. This result partly explains why the VM startup for Windows machines is much longer than the Linux machines (there are booting time differences as well). Moreover, this result can also be used to calculate the internal data transfer rate between the VM instances and the image store for each cloud provider. The result is shown in TABLE IV. Rackspace now has the fastest OS image downloading speed (22.5 MB/s). The data transfer rates for EC2 and Rackspace are consistent with their cloud storage performances in other studies [17][14]. However, Azure shows an order of magnitude slower performance compared to EC2 and Rackspace, which is also inconsistent with the data transfer rate of its blob storage. Currently, we do not have an explanation for this fact. In practice, VM image size is one of the few factors that can be controlled by the cloud users and it can largely affect the speed of the VM startup process as shown in this experiment. Thus, carefully planning the software stack installed on the OS image and removing unnecessary files is a way to keep the image small and speed up the VM acquisition process. Sometimes, it is a tradeoff decision that needs to be made between the VM startup time and the first-time service loading delay.

TABLE IV. DATA TRANSFER RATE BETWEEN VMS & IMAGE STORE

Cloud	Average Data Transfer Rate between VM and Image Store
EC2	10.9 MB/s
Azure	1.1MB/s
Rackspace	22.5 MB/s

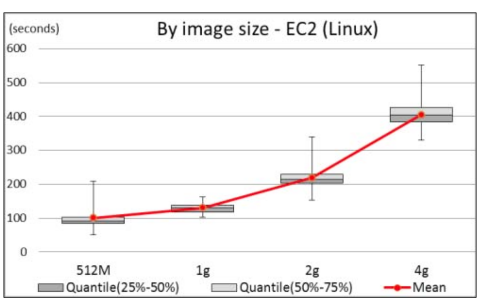

Figure 5. EC2 VM startup time – by image size.

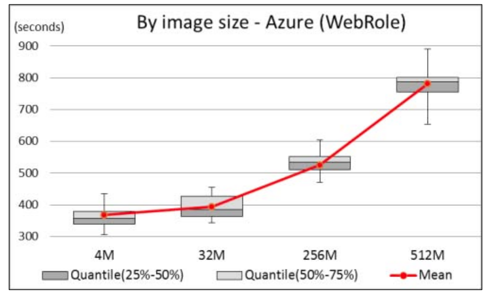

Figure 6. Azure VM startup time – by image size.

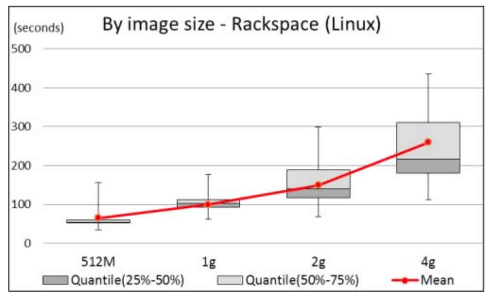

Figure 7. Rackspace VM startup time – by image size.

C. *By VM Instance Type*

The third experiment focuses on the relationship between the VM startup time and the VM instance type. Currently, cloud providers offer different instance types to meet user computing needs. For example, EC2 offers twelve types of instances, Azure offers five types of instances and Rackspace offers seven types of instances. We summarize the instance types that are included in this experiment in the table below.

TABLE V. VM TYPES FOR EC2, AZURE & RACKSPACE

Type	CPU	Mem	Disk
Amazon EC2 (1 CU = 1.0-1.2 GHz 2007 Opteron**)**			
t1.micro	up to 2 CU	613 MB	EBS
m1.small	1 CU	1.7 GB	160 GB
m1.large	4 CU	7.5 GB	850 GB
m1.xlarge	8 CU	15 GB	1690 GB
m2.xlarge	6.5 CU	17.1 GB	420 GB
c1.medium	5 CU	1.7 GB	350 GB
Windows Azure			
ExtraSmall	1.0 GHz	768 MB	20 GB
Small	1.6 GHz	1.75 GB	225 GB
Medium	2*1.6 GHz	3.5 GB	490 GB
Large	4*1.6 GHz	7 GB	1000 GB
ExtraLarge	8*1.6 GHz	14 GB	2040 GB
Rackspace			
Type I	N/A	256 MB	10 GB
Type II	N/A	512 MB	20 GB
Type III	N/A	1024 MB	40 GB
Type IV	N/A	2048 MB	80 GB
Type V	N/A	4096 MB	160 GB
Type VI	N/A	8192 MB	320 GB
Type VII	N/A	15872 MB	620 GB

From Figure 8. we do not see any clear patterns based on the instance types in EC2. A larger instance type does not necessarily imply a longer VM startup time. For example, for m1.small, m1.large and m1.xlarge, all of their VM startup time are around 100 seconds. However, Azure and Rackspace clearly show that the VM startup time increases as the instance type goes larger. A notable difference is that in Azure and Rackspace, the configuration of next larger instance type is always twice more powerful (faster or larger) than the previous instance type, which may take longer to allocate the resources in the cloud. In EC2, this is not the case. This could explain why EC2 does not show the linearly (roughly) increasing VM startup time as Azure and Rackspace.

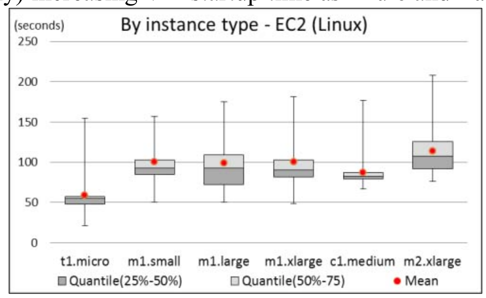

Figure 8. EC2 VM startup time – by instance type.

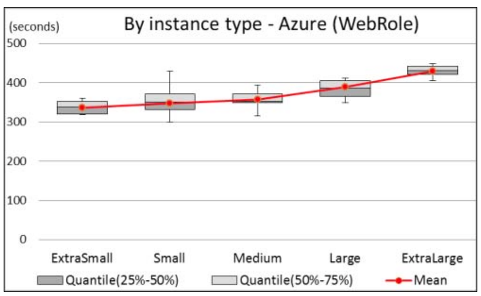

Figure 9. Azure VM startup time – by instance type.

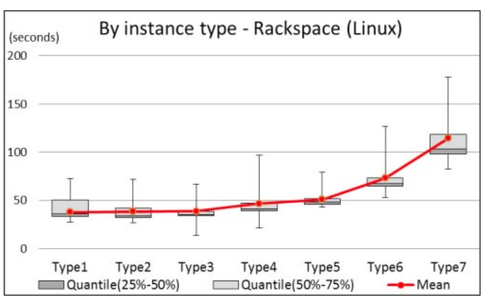

Figure 10. Rackspace VM startup time – by instance type.

D. By VM Location

In this experiment, we collect the cloud VM startup time by different locations. For high service availability and to reduce the communication latency of geographically distributed cloud users, cloud providers offer several locations to provision user requested VM instances. TABLE VI. summarizes the data center locations of EC2 and Azure. Currently, Rackspace does not support the VM-location option and we do not include Rackspace in this experiment. Figure 11. shows the VM startup time for m1.small instances across different data centers in EC2. In fact, during our experiment, EC2 announced two new regions. They are the US West region in Oregon on 9th Nov, 2011 and the South America region in Sao Paulo on 15th Dec, 2011. Considering the number of data points we can collect, we include the Oregon data center but not the Sao Paulo data center in this experiment. All data centers show similar performances (98 seconds) except that the newly established data center shows slightly slower VM startup time (120 seconds) and greater variance. This result contradicts the impression that a newly established data center may have lower workload and therefore shorter VM startup time than the existing data centers. Figure 12. shows the VM startup time for small WebRole applications in Azure. Azure also shows similar performances across different data center locations. Therefore, we conclude that currently the location is not a significant factor contributing to the duration of the VM startup process in the cloud. However, users need to consider the longer VM startup time in newly estalished data centers.

TABLE VI. VM LOCATIONS FOR EC2 & AZURE

Regions	Location
Amazon EC2	
US East (us-east-1a,1b,1d)	Virginia, US
US West (us-west-1a,1b,1c)	California, US
US West* (us-west-2a,2b)	Oregon, US
EU West (eu-west-1a,1b,1c)	Ireland, EU
Asia Pacific (ap-southwest-1a,1b)	Singapore
Asia Pacific (ap-northeast-1a,1b)	Tokyo, Japan
Windows Azure	
North-central US	Chicago, IL
South-central US	San Antonio, TX
East Asia	Hong Kong, China
South East Asia	Singapore
West Europe	Amsterdam, Netherland
North Europe	Dublin, Ireland

*announced on 9th Nov, 2011

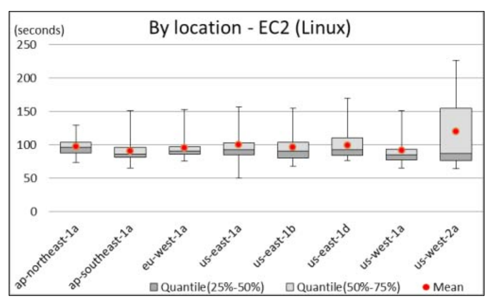

Figure 11. EC2 VM startup time – by location.

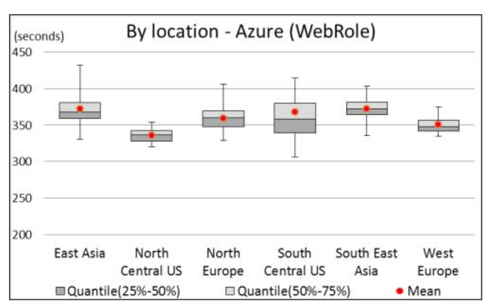

Figure 12. Azure VM startup time – by location.

E. Multiple Instances

The above results show the VM startup time for a single-instance request in the cloud. In this experiment, we will show the VM startup time for a multiple-instance acquisition request. By default, EC2 allows cloud users to run 20 on-demand instances per region [12]. Azure allows 20 VM cores per account [13]. Currently, Rackspace only allows users to acquire one VM instance per request [18]. In other words, cloud users have to repeat the single-instance request for multiple times if they need more than one instance. Therefore, we study the VM startup time of a 16-instance request only for EC2 and Azure. The results are shown in Figure 13. and Figure 14. For EC2, we do not see the VM startup time increases as the number of acquired instances increases. All instances have the VM startup time around 100 seconds. However the variance becomes greater compared to single-instance requests. In Azure, the startup time increases linearly as the number of instance increases. For a 16-instance acquisition request, the average VM startup time of the 16^{th} instance is 110 seconds slower than the 1^{st} instance (from 395 sec to 505 sec). In other words, when a large number of instances are acquired at the same time, the amortized average VM startup time becomes longer. Therefore, users need a longer preparation time than single-instance requests. Also note that the last several VMs have greater variance compared to the early instances.

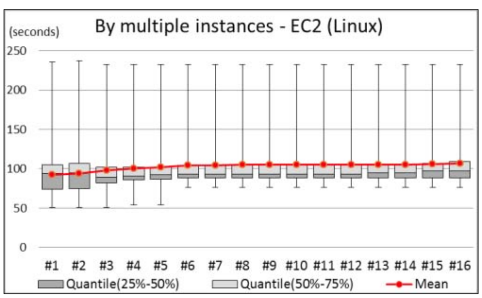

Figure 13. EC2 VM startup time – by multiple instances.

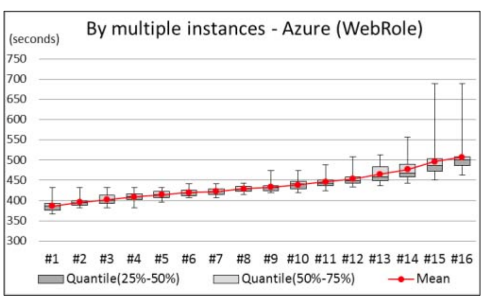

Figure 14. Azure VM startup time – by multiple instances.

F. Spot Instances

Finally, we collect the VM startup time for spot instances (SI). Currently, EC2 offers spot instances to allow users to bid for their unused capacity and run those instances for as long as their bid exceeds the current spot price. The spot price changes periodically based on supply and demand, and customers whose bids exceed it gain access to the available spot instances. Spot instances have different life-cycle statuses compared to the on-demand instances (ODI). Instead of the "pending" status, they now have "open" and "active". "Open" means the spot instance request is waiting to be served, while "active" means the acquired VM is being provisioned, which is just like the "pending" status for the on-demand instances. Based on our experiment results, we can see the "open to active" phase is the most time consuming part in the acquisition process of spot instances. In other words, spot instance requests take significantly longer time waiting to be served than the on-demand instances. In this section, we report both the quantiles and the average for the spot instances to be active (ActiveQuantile and ActiveMean in the figures) and only the average for the whole VM startup process (VM Startup Mean). For both spot and on-demand instances, the time from being provisioned (active or pending) to the first successful login is consistent. It is around 95 seconds for m1.small instances. This consistency also holds for different instance types and data center locations.

Time of the day (Figure 15.) – Similar to ODI, SI does not show any clear pattern based on time of the day. However, the average VM startup time has been increased to 519 seconds, in which 423 seconds are spent in waiting for the requests to be served. SI also shows greater variance compared to ODI. ***Instance type*** (Figure 16.) – Similar to ODI, the VM startup time of SI does not show significant differences based on the instance type. One notable difference is that t1.micro now has similar startup time to other larger instance types. ***Location*** (Figure 17.) – The VM startup time of SI shows greater differences based on the locations. The US East region (us-east-1a, b, d) has longer (2 or 3 times) VM startup time than the other locations. We believe this change implies the levels of the SI workload in different regions. This fact means that cloud users can choose faster locations to launch their spot instances to reduce the waiting time by 7 minutes. ***Multiple instances*** (Figure 18.) – SI also scales well with multiple instances. The "active" difference between the 16^{th} and the 1^{st} instances is within 23 seconds, while the real VM provisioning time is consistent across all the 16 instances, which is around 96.9 seconds. ***Real-time price*** (Figure 19.) – The VM startup time of SI does not show a significant correlation with the spot prices. A higher spot price does not necessarily imply a longer VM startup time. For all the seven real-time prices we observed, the differences of the VM startup time are within 96 seconds.

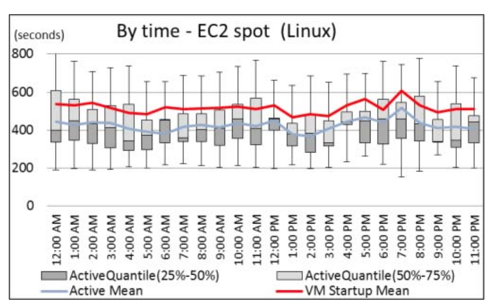

Figure 15. EC2 VM startup time (SI) – by time.

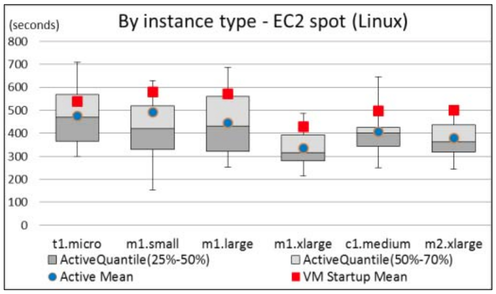

Figure 16. EC2 VM startup time (SI) – by instance type.

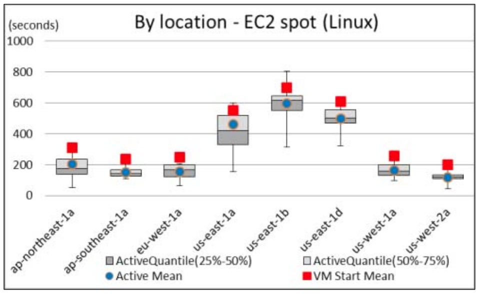

Figure 17. EC2 VM startup time (SI) – by location.

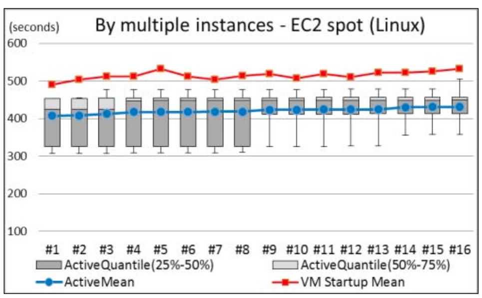

Figure 18. EC2 VM startup time (SI) – by multiple instances.

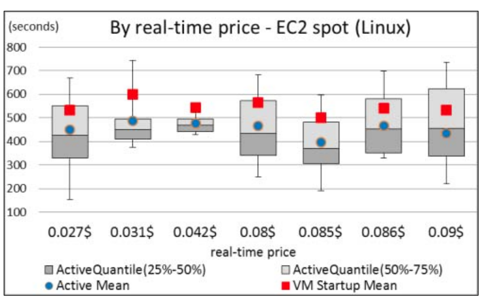

Figure 19. EC2 VM startup time (SI) – by price.

G. Other Facts

In this section, we report some other facts that can affect the VM acquisition decisions. Particularly, we will talk about (1) the prices of spot instances, (2) the success rate of the instance acquisition requests and (3) the average time to release a cloud VM instance.

The original idea of spot instances is that EC2 can offer cloud users unused capacity to improve their resource utilization and cloud users could purchase computing resources at a lower price to reduce cost. From the economic point of view, spot instances should be cheaper than on-demand instances because their life-cycles are not fully controlled by the cloud users and they take longer time to startup. However, we find this is not always the case. Figure 20. shows the percentage that the prices of SI stay below, stay the same and go above the ODI in our experiment period. For t1.micro instances, there is 53% of the time that spot prices are higher than or equal to the on-demand instances. For other instance types, this percentage is between 3% and 21%. This fact means that even if users bid SI with on-demand prices, the VMs may still be not available. The VM startup time is then dependent on the movement of spot price. One possible solution is that users can actually bid a higher price than ODI, but make sure the average cost in the long run is still cheaper than ODI. In this case, users sacrifice some money for shorter (more certain) VM startup time and reduce the possibility of instance shut-downs by EC2.

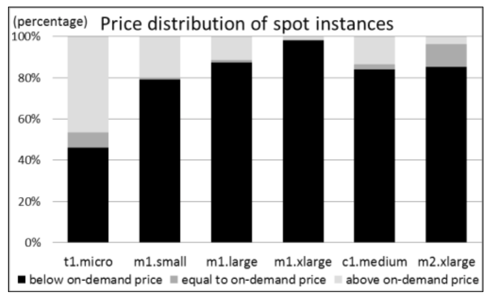

Figure 20. Price distribution of spot instances

VM acquisition requests are not always successfully served by the cloud providers. Sometimes, the acquired instances could go to error states. For example, the OS cannot be booted or the instances cannot be connected. We report the success rate of the instance acquisition requests in TABLE VII. Both EC2 and Azure have a success rate above 99%, however, Rackspace only shows a 92% success rate. This fact means that, users need to check the VM status after each instance acquisition request. When unhealthy instances are provisioned, it means the acquired VMs will take forever to startup. Cloud users need to manually delete the resource and reacquire the instances. This is one factor in the VM startup process that cannot be controlled by the users.

In addition to studying the VM acquisition time, we also record the VM release time. The VM release time is much shorter and more stable than the VM acquisition time. We don't see any significant differences by different factors. Therefore, we only report the average VM release time for each cloud provider. In the table below, we use the duration from the time of issuing the instance deletion requests to the time that the VMs cannot be successfully pinged as the VM release time. Azure shows a slightly longer VM release time than EC2 and Rackspace.

TABLE VII. THE ACQUISITION SUCCESS RATE & VM RELEASE TIME

Cloud	Success Rate	VM Release Time
EC2	99.2%	3-8 seconds
Azure	99.6%	8-21 seconds
Rackspace	92.0%	3-8 seconds

IV. RELATED WORK

Though the performance of the VM startup process has been mentioned in several research papers, they just served as part of the work to evaluate the overall performance for a cloud provider. To the best of our knowledge, there has not been any complete and systematic performance study on the VM startup time based on related factors as we did in this paper. For example, in [10], the authors compared the VM startup time between single- and multiple-instance requests in EC2. [11] analyzed the startup time for WebRole and WorkRole in Azure. Neither of them considers the factors like time of the day, OS image size, instance type and data center locations. They do not consider the newly published services such as spot instances and VMRole as well. From these studies conducted two years ago, we can see Azure shows a 200 second improvement and smaller variance on the VM startup time while EC2's performance does not change.

People are aware of this waiting time of the dynamically provisioned resources in the cloud and have developed techniques to speed up the VM provisioning process. For example, [19] designed a fast cloud deployment framework based on VM cloning. [20] developed a fast start technique by restoring previously created VM snapshots of fully initialized application. [21] designed a VM image format called FVD to support instant VM creation and migration. [22] proposed a chuck-level virtual machine image distribution network to enable collaborative sharing in cloud data centers and reduce the VM instance provisioning time.

In addition, researchers are particularly interested in using spot instances to lower their job execution cost and accelerate job execution speed [3][4][5][6]. Their works are based on the assumptions that spot instances are cheaper than on-demand instances. Therefore, cloud users can acquire more computing power using the same amount of money or save more money for the same amount of workload than on-demand instances. However they do not explicitly consider the longer VM startup time and the higher prices of spot instances in their models. The spot instances may not always be a better candidate than the on-demand instances.

V. CONCLUSION & FUTURE WORK

One of many advantages of cloud computing is elasticity, the ability to scale up and down in response to demand dynamically. Understanding the VM startup time is important for time-critical applications and cloud auto-scaling mechanisms. In this paper, we analyzed the relationship between the VM startup time and different factors, extract useful information, compare the performance across three cloud providers and make recommendations whenever possible. In the future, we plan to extend this performance study to more cloud providers and instance types, and publish the updated performance results periodically.

REFERENCES

[1] Are Long VM Instance Spin-Up Times In The Cloud Costing You Money? http://highscalability.com/blog/2011/3/17/are-long-vm-instance-spin-up-times-in-the-cloud-costing-you.html

[2] Why does Azure deployment take so long? http://stackoverflow.com /questions/5080445/why-does-azure-deployment-take-so-long

[3] S. Yi, D. Kondo, and A. Andrzejak, "Reducing Costs of Spot Instances via Checkpointing in the Amazon Elastic Compute Cloud," In The 3rd IEEE International Conference on Cloud Computing (CLOUD 2010), Miami, FL USA, 2010.

[4] S. Wee, "Debunking Real-Time Pricing in Cloud Computing," In The 11th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid 2011), Newport Beach, CA, USA, 2011.

[5] N. Chohan, C. Castillo, M. Spreitzer, M. Steinder, A. Tantawi, and C. Krintz. "See Spot Run: Using Spot Instances for Mapreduce Workows," In Proceedings of the 2nd USENIX conference on Hot topics in cloud computing (HotCloud 2010), pages 7-7, Berkeley, CA, USA, 2010.

[6] M. Mattess, C. Vecchiola, and R. Buyya, "Managing Peak Loads by Leasing Cloud Infrastructure Services from a Spot Market," In Proceedings of the 2010 IEEE 12th International Conference on High Performance Computing and Communications (HPCC 2010), pages 180-188, Washington, DC, USA, 2010.

[7] M. Mao, Jie Li and M. Humphrey, "Cloud Auto-Scaling with Deadline and Budget Constraints," In Proceedings of 11th ACM/IEEE International Conference on Grid Computing (Grid 2010), Brussels, Belgium, Oct 25-28, 2010.

[8] M. Mao and M. Humphrey, "Auto-Scaling to Minimize Cost and Meet Application Deadlines in Cloud Workflows," In Proceedings of 2011 International Conference for High Performance Computing, Networking, Storage and Analysis (SC 2011), Seattle, WA, Nov 15-20, 2011.

[9] Joe Weinman, Time is Money: The Value of "On-Demand", http://www.joeweinman.com/Resources/Joe_Weinman_Time_Is_Money.pdf

[10] S. Ostermann, A. Iosup, N. Yigitbasi, R. Prodan, T. Fahringer and D. Epema, "A Performance Analysis of EC2 Cloud Computing Services for Scientific Computing," CloudComp 2009.

[11] Z. Hill, M. Mao, J. Li, A. Ruiz-Alvarez, and M. Humphrey, "Early Observations on the Performance of Windows Azure," in 1st Workshop on Scientific Cloud Computing (ScienceCloud 2010), 2010.

[12] Amazon EC2. http://aws.amazon.com/ec2/

[13] Windows Azure. http://www.microsoft.com/windowsazure/

[14] Rackspace. http://www.rackspace.com/

[15] HostMonk, All Rankings of Cloud Providers. http://www.hostmonk.com/ranks/cloud

[16] SearchCloudComputing. Top 10 Cloud Computing Providers of 2011. http://searchcloudcomputing.techtarget.com/feature/Top-10-cloud-computing-providers-of-2011

[17] EC2 Data Transfer Rate. http://blog.rightscale.com/2007/10/28 /network-performance-within-amazon-ec2-and-to-amazon-s3/

[18] Rackspace Cloud Server Limits. http://docs.rackspace.com /servers/api/v1.0/cs-devguide/content/Limits-d1e997.html

[19] X. Wu, Z. Shen, R. W and Y. Lin, "Jump-start cloud: efficient deployment framework for large-scale cloud applications," In Proceedings of the 7th International Conference on Distributed Computing and Internet Technology (ICDCIT 11).

[20] J. Zhu, Z. Jiang and Z. Xiao, "Twinkle: A Fast Resource Provisioning Mechanism for Internet Services," In The 30th IEEE International Conference on Computer Communications (INFOCOM 2011), Shanghai, China 2011.

[21] C. Tang, "FVD: A High-Performance Virtual Machine Image Format For Cloud," In Proceedings of the 2011 USENIX Conference on USENIX Annual Technical Conference (USENIXATC'11).

[22] C. Peng, M. Kim, Z. Zhang and H. Lei, "VDN: Virtual Machine Image Distribution Network for Cloud Data Centers," In The 31th IEEE International Conference on Computer Communications (INFOCOM 2012), Orlando, FL USA, 2012.