TECHNICAL WHITE PAPER – **OCTOBER 2017**

VMWARE HORIZON CLOUD SERVICE ON MICROSOFT AZURE

RDS Desktop and Application Scalability

vmware®

Table of Contents

Executive Summary

This white paper provides analysis into the achievable scalability and optimal user densities for VMware Horizon Cloud Service™ on Microsoft Azure, along with providing some cost considerations for deployment at scale.

Horizon Cloud Service on Microsoft Azure provides a single platform for delivering virtualized Windows applications and shared desktop sessions from Windows Server instances using Microsoft Remote Desktop Services (RDS) running in Microsoft Azure. With Horizon Cloud, you can publish business-critical Windows apps alongside SaaS and mobile apps and desktops in a single digital workspace, easily accessed with single sign-on from any authenticated device or OS.

This white paper describes the use of the platform to meet key business requirements such as making standard Windows applications available to employees, and targets use cases such as task workers and knowledge workers. We carried out extensive testing to evaluate the performance and capacity characteristics of VMware Horizon Cloud Service Apps and Desktops in a Microsoft Azure environment. This paper describes user densities we qualified for differing workloads, as summarized in Table 1 and Figure 1:

SERVER MODEL	KNOWLEDGE WORKER CONCURRENT SESSIONS	TASK WORKER CONCURRENT SESSIONS
Small (D2v2)	20	26
Medium (D3v2)	30	50
Large (D4v2)	60	85
GPU (NV6)	20[1]	n/a

Table 1: Use Case Concurrent Sessions

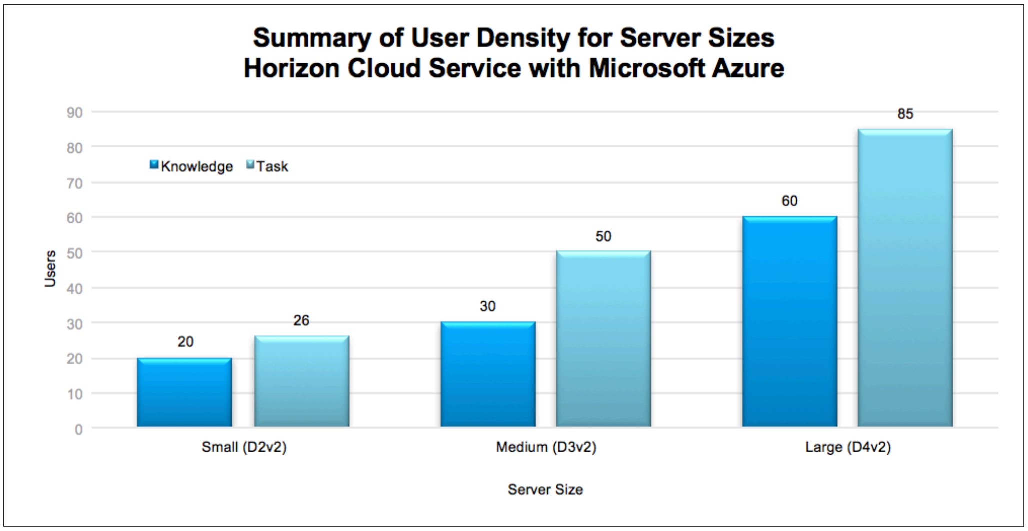


Figure 1: Summary of Results

[1] NV6 currently limited to 20 sessions maximum on Windows Server 2012 R2, and not yet supported on Windows Server 2016..

Power management is then detailed, and the analysis demonstrates how the power management algorithms in Horizon Cloud Service can greatly save an enterprise a significant amount of money by powering down servers when they are not required, and consideration is given to selecting more ‘small’ servers rather than fewer ‘large’ servers, again to aid in cost management.

Audience

This document is intended for IT architects and administrators who want to understand the performance and scale attributes of the VMware Horizon Cloud Service on Microsoft Azure platform in a virtualized RDSH environment. The reader should have a solid understanding of desktop and application virtualization, and familiarity with VMware Horizon®, VMware Horizon Cloud Service, and Microsoft Azure, in addition to an understanding of sizing and performance concepts.

Horizon Cloud Service on Microsoft Azure Overview

Horizon Cloud is a software service from VMware that allows customers to easily and cost-effectively deploy cloud-hosted or on-premises virtual desktops and apps to any device, anywhere. As more and more organizations adopt a multi-cloud strategy to leverage "best-fit" cloud capabilities, avoid cloud vulnerability, and optimize cost, they face new challenges of moving between clouds and inefficiencies of operating multiple solutions.

VMware has recently launched an extension to Horizon Cloud Service that leverages Microsoft Azure for desktop and application capacity. Horizon Cloud Service on Microsoft Azure is designed to help organizations transform Windows applications into a software-as-a-service (SaaS) model. This helps IT teams eliminate the burden of managing physical infrastructure and move to SaaS as part of the journey to a unified digital workspace.

This white paper identifies the user density scalability tests that have been conducted by VMware engineers using Horizon Cloud Service on Microsoft Azure. In addition, the white paper aims to identify some of the cost considerations so that a well-sized environment can be selected to meet the enterprise's needs, delight end users, and limit spend.

Introduction to Microsoft Azure

Microsoft Azure is a very flexible and scalable cloud platform offering high reliability across its more than 40 data centers globally. It allows customers to deploy and manage infrastructure easily across a global footprint.

RDSH Features

Horizon Cloud provides a simple way from any browser to create and manage a RDSH environment running on Microsoft Azure.

The published-applications feature supports a wealth of remote-experience features. These include everything from the HTML Access web client to client-drive redirection, access to locally connected USB devices, file-type association, Windows media redirection, content redirection, printer redirection, location-based printing, 3D rendering, smart card authentication, and more. The published-applications feature can leverage the Blast Extreme Adaptive Transport (BEAT) and PCoIP display protocols from VMware, providing a rich user experience using zero, thin, laptop, PC, or mobile clients over LAN, WAN, or bandwidth-limited connections.

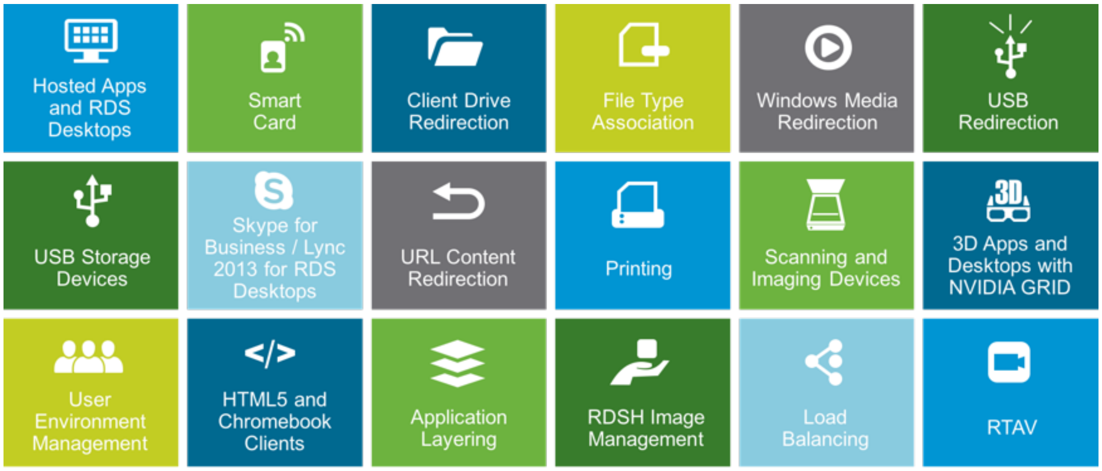

Hosted Apps and RDS Desktops
Smart Card
Client Drive Redirection
File Type Association
Windows Media Redirection
USB Redirection
USB Storage Devices
Skype for Business / Lync 2013 for RDS Desktops
URL Content Redirection
Printing
Scanning and Imaging Devices
3D Apps and Desktops with NVIDIA GRID
User Environment Management
HTML5 and Chromebook Clients
Application Layering
RDSH Image Management
Load Balancing
RTAV

Figure 2: Remote-Experience Features Available with Published Applications

With published applications, you install applications on servers with the Microsoft Remote Desktop Session Host (RDSH) role, and entitle applications to corporate users through the Horizon Cloud Administration console. Once authenticated, users can launch an application, save files, and use network resources from a remote RDSH server—just as if the users had the application installed on their local computer, tablet, or phone.

Publishing applications using the published-applications feature simplifies management of line-of-business applications, allows the delivery of Windows applications to non-Windows devices, and can potentially provide licensing advantages. This strategy can reduce CapEx and OpEx costs, and simplify installation, upgrades, and troubleshooting.

End users launch VMware Horizon Client™ or the HTML Access web client, and log in to the server that brokers connections to published apps. Users then see a catalog of published apps, as well as session-based or single-user virtual desktops, if desktops have been configured.

Figure 3 details the components of a Horizon Cloud Service on Microsoft Azure architecture:

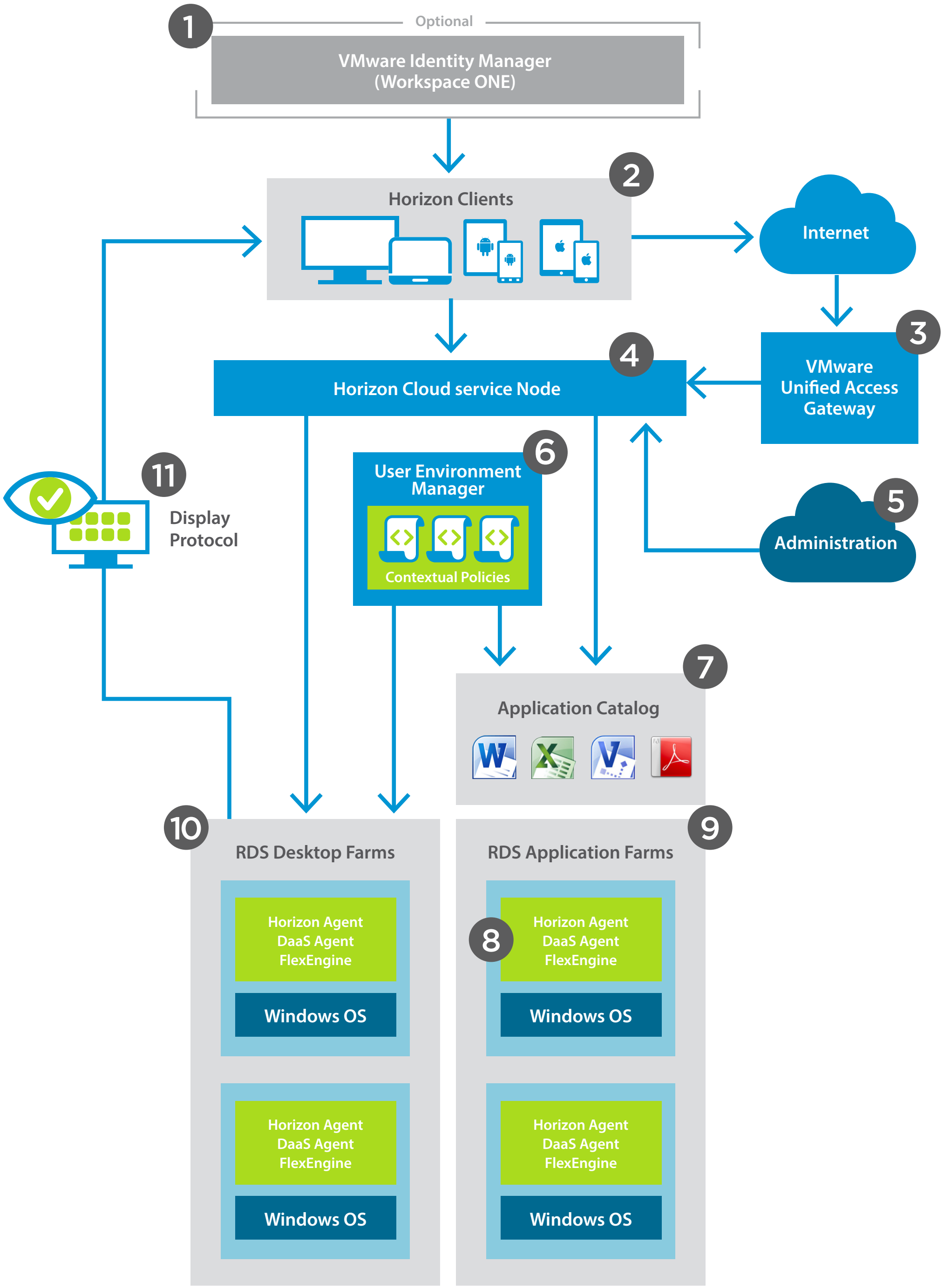


Figure 3: Components of Horizon Cloud Service on Microsoft Azure

Horizon Cloud Service on Microsoft Azure includes RDSH-based published applications and shared session-based virtual desktops.

1. **VMware Identity Manager™** (VMware Workspace™ ONE™) [Optional] – Provides a convenient way to manage all SaaS application entitlements, and desktop and app entitlements, from a convenient portal.
2. **Horizon Clients** – Client software is available from app stores or from VMware for iOS, Android, Chrome, Windows, Linux, and macOS so that users can access published applications and desktops from any device. An HTML Access web client is also available, and it does not require installing any software on client devices.
3. **VMware Unified Access Gateway™** [Optional] – Provides secure access from the WAN without needing to route user sessions across a VPN/ExpressRoute connection.
4. **Horizon Cloud node** – Horizon Client is configured to access the Horizon Cloud node that is deployed in Microsoft Azure. This server, which integrates with Windows Active Directory, provides access to virtual desktops and published applications.
5. **Horizon Cloud Administration Console** – All management operations are performed from a single administrative graphical user interface—from initial build-out and configuration of the environment, to image creation and farm configuration and entitlements. It also provides a convenient management portal for ongoing monitoring and reporting purposes.
6. **VMware User Environment Manager™** [Optional] – Offers the capability to provide a user experience customized to each user's preferences, enabling each user to have their own desktop or app look, feel, and functionality.
7. **Application catalog** – Each application that you select to publish becomes an application definition. For example, using the New Application wizard, if you select the Paint and Calculator apps to publish, when you complete the wizard, you will have a Paint application and a Calculator application definition. These applications can then be entitled to users in an Application Assignment. Assignments can include apps from multiple farms on the same Horizon Cloud node.
8. **Agents** – Horizon Cloud automatically installs the Horizon Agent and the DaaS agents, and optionally you can manually install the User Environment Manager FlexEngine service on the master images for Microsoft RDSH servers.

   The Horizon Agent communicates with Horizon Client to provide features such as connection monitoring, virtual printing, folder sharing (client-drive redirection), and access to locally connected USB devices.

   FlexEngine [Optional] – FlexEngine is the User Environment Manager agent. FlexEngine starts at login and imports policy settings, including application and user environment settings, from a configuration share. This agent also optionally loads personalization settings from a user profile archives share. You use the provided Group Policy Object (GPO) administrative templates (`.admx` files) to enable and configure FlexEngine.
9. **RDSH application farms and VMs** – The server VMs are grouped (and managed by) RDSH farms. One or more RDSH servers make up a farm, and from that farm you create application and shared session-based desktop pools. Once assignments have been created, end users can access desktops and apps from any device, anywhere.
10. **RDSH desktop farms and VMs** – These farms are similar to the RDSH application farms above, but provide session-based desktops, where several users share the same server to deliver a rich desktop experience.

11. **Display Protocol** – BEAT technology provides a highly optimized display protocol that works over LAN, WAN, and highly lossy mobile networks, while still delivering a fantastic user experience. Blast Extreme, HTML Access, and PCoIP are also supported.

The key challenge to system architects when designing such an environment, is to identify how many users can share a single RDS server in a farm to ensure that they get a good user experience, while maximizing the number of users on a server to help minimize costs. No two users are the same, and no two enterprises are the same—everyone will use the system in different ways. This paper aims to test the user density in a fair comparative way.

Scalability Testing of Horizon Cloud Service on Microsoft Azure Instances

Microsoft Azure provides a number of VM sizes (see Sizes for Windows virtual machines in Azure for details). Horizon Cloud has selected the Dv2 series family of VMs for use with RDS servers, due to their CPU/memory ratios, and from extensive testing on the VMware Horizon 7 product line, which recommended similar specified machines.

To provide perspective on the scalability of Horizon Cloud Service on Microsoft Azure, VMware engineers conducted a number of tests on various Microsoft Azure Dv2 series VM instances. Table 2 provides an overview of the three VM instances that were tested.

SERVER SIZE	AZURE INSTANCE TYPE	VCPU	MEMORY GB	TEMPORARY STORAGE (SSD) GB
Small	Standard D2v2	2	7	100
Medium	Standard D3v2	4	14	200
Large	Standard D4v2	8	28	400
GPU	NV6 (half M60)	6	56	380

Table 2: Specifications of Tested VM Instances

Note that due to an NVIDIA driver limitation on Windows Server 2012 R2, the maximum number of sessions is limited to 20 (any additional sessions connected after this result in a black screen). Due to a driver limitation in Horizon 7.3.1 Agent, Windows Server 2016 is not yet supported.

Each of the three VM instances had the following software packages installed:

- Windows Server 2012 R2 Datacenter
- VMware Horizon DaaS® Agent 17.2.0
- VMware Horizon Agent 7.3.1
- VMware Desktop RAWC Workload Simulator 2.0.0
- Microsoft Office Professional Plus 2016 16.0.4266.1001
- Adobe Reader XI 11.0.10
- Google Chrome 60.0.3112.113

We elected to run all of the tests using the Blast Extreme Adaptive Transport (BEAT), however we also ran some comparative tests with the PCoIP protocol that produced similar results.

Test Method

The Reference Architecture Workload Simulator (RAWC) was developed by VMware and used to conduct our performance and scalability tests. RAWC is used for similar profiling on VMware Horizon 7 (on-premises product) workloads as well as workloads for Horizon Cloud. RAWC generates a realistic, adjustable workload with various applications and resides on the VMs under test. In addition to the workload that resides on the VM, RAWC contains two additional components: a Controller and one or more Session Launchers that manage session scale.

The RAWC Controller and Session Launcher reside on separate Windows Server 2012 R2 VMs. Sessions are launched using the specified display protocol into the RDS farms running on Microsoft Azure. Sessions are launched every 30 seconds. Test data was gathered once a steady state of user sessions was established.

The RAWC workload launches the applications in random order. RAWC captures the open and close times for the applications in the timing logs. RAWC charting software is used to consolidate the logs and calculate the average open times for the applications. In addition, CPU, memory, and disk IOPS values are captured.

Microsoft Office 2016 was installed for this testing. During the test, RAWC creates a Word document, types text into that document, and then saves it. At that point, the Word document is minimized to the taskbar and then reopened again later in the test iteration for more typing of text before RAWC does a final save and then closes the document. RAWC performs similar activities with Excel. A PowerPoint presentation including text and graphics is opened and RAWC runs through a full-screen slide show before closing the presentation.

Adobe Reader XI was also installed. Two Adobe PDF documents are provided. The first Adobe PDF document is opened and random pages are browsed. This browsing occurs twice during an iteration, and the document is also minimized to the taskbar. The second Adobe PDF document is opened and page scrolling is performed. Based on the test configuration, you can specify the scroll speed from very slow to very fast, as well as determine the amount of time you want RAWC to scroll through the PDF document. Both documents are reopened from the taskbar before being closed at the end of the iteration.

Internet Explorer and Google Chrome open to a home page on the Internet and are minimized to the task bar and reopened before closing at the end of the iteration.

The activities described occur in each iteration of the test. In addition to defining the scroll speed and time for a PDF document, RAWC provides additional tuning parameters such as a random think time (time between application launches and use) and typing speed. Depending on the values, these parameters can increase or decrease the load on the server. The closer together the applications perform tasks, the more load is placed on the server.

Table 3 provides information about the infrastructure VMs that were set up on Azure instances and used during the testing cycle.

DESCRIPTION	QUANTITY	AZURE INSTANCE	SOFTWARE INSTALLED
RAWC Controller	1	Standard DS11 v2 (2 vCPUs, 14 GB)	• Windows Server 2012 R2 • Desktop RAWC Controller 2.0
RAWC Session Launcher	2	Standard DS13 v2 (8 vCPUs, 56 GB)	• Windows Server 2012 R2 • Desktop RAWC Session Launcher 2.0 • VMware Horizon Client 4.4.0.5856
Active Directory controller and DNS server	1	Standard A1 (1 vCPU, 1.75 GB)	• Windows Server 2012 R2

Table 3: Azure Instance Specifications

Depending on the number of user sessions, RAWC was configured to run three to five iterations. Tests lasted from 50 to 60 minutes for three iterations and 1.5 to 1.75 hours for five iterations. The higher user densities required additional time to allow for 15–25 minutes of steady-state testing. Steady state occurs after all the sessions have been launched and the users have logged in, and before the first user finishes the test and logs out. At least five test runs were performed at each of the user densities and then averaged.

Tests were performed over a 2-month period between July and September 2017.

Because the servers are running on a public cloud environment, the actual performance obtained will vary based on location and loading on the underlying physical hardware, which is outside of the control of VMware. As such, it was possible to see a variation in the results obtained, and so it is advisable to perform your own user-acceptance testing over a period of time for any user density value selected.

For each user density, we observe the following metrics:

- Application Pool Open times
- CPU Peak Utilization, and Average Steady State CPU Utilization
- Steady State Total Disk Read and Write Operations (IOPSs)

Steady-state values are measured once all users are logged in and the test workloads are being operated. These measurements are averaged, as well as a peak value measured during this time period.

For Application Pool Open times, we aim to find a user density that keeps this less than 4 seconds. We also look for any inflection point where this starts to ramp up much faster.

For CPU Peak Utilization and Average Steady State CPU Utilization, we look for these values flat-lining, or maxing out above 90 percent.

For Steady State Total Disk Read and Write Operations (IOPSs), again, we look for any inflection points.

Test Scenarios

Two different test scenarios were conducted to simulate different use-case scenarios.

Scenario #1 – Knowledge Worker

The following applications were used to simulate a Knowledge Worker workload: Microsoft Office 2016 Word, Excel, and PowerPoint; Adobe Reader XI to browse random PDF pages, and scroll a PDF document; Internet Explorer; and Google Chrome.

For Knowledge Worker, additional configuration settings include:

- Think time: 30 seconds – This is the upper limit of a random delay calculation that is performed between applications. The lower limit is 1/10 of a second.
- Scroll speed: 5 – This is a medium speed. Scroll speed values are 1 through 9.
- Scroll time: 30 seconds. – This task is performed twice in each iteration of the test.
- PDF browse: 1 page. A random page is browsed. This task is performed twice in each iteration of the test.
- Typing speed: 5000 – This speed is moderately fast with minimal breaks between sentences.

Scenario #2 – Task Worker

The Task Worker scenario is a lighter workload than the Knowledge Worker one. As a result, it is expected that you can have more task worker sessions per server compared to knowledge workers on a similarly sized server.

The following applications were used to simulate a Task Worker workload: Microsoft Office 2016 Word and Excel, Adobe XI to browse random PDF pages, and Google Chrome.

We repeated the tests for different user densities on each of the server models (Small, Medium, Large) for both Knowledge Worker and Task Worker scenarios.

For Task Worker, additional configuration settings include:

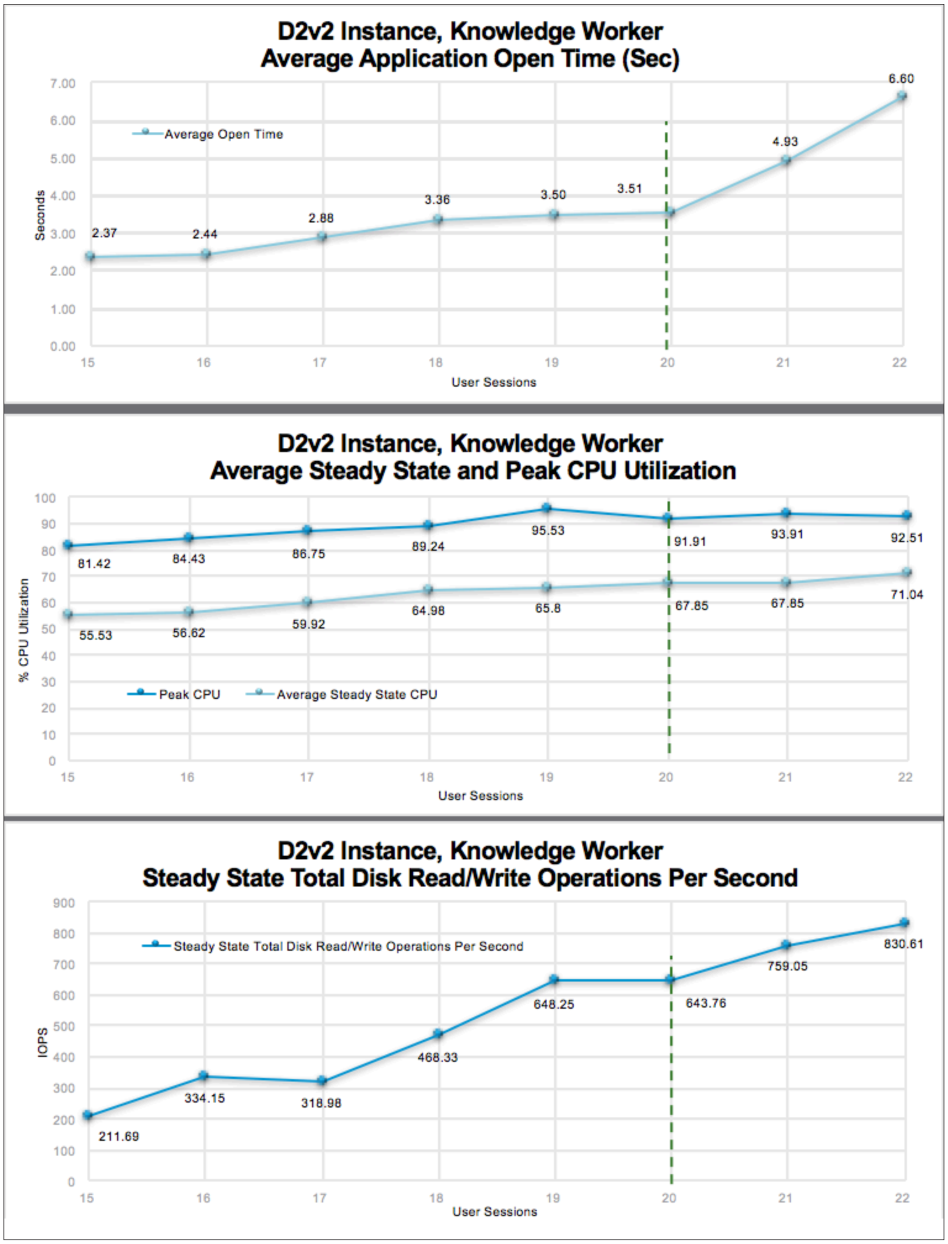
- Think time: 60 seconds. – This is the upper limit of a random delay calculation that is performed between applications. The lower limit is 1/10 of a second.
- The Task Worker workload did not scroll the PDF document.
- PDF Browse: 1 page. A random page is browsed. This task is performed twice in each iteration of the test.
- Typing speed: 5000 – This speed is moderately fast with minimal breaks between sentences.

Test Results

The following test results show the data measured for each of the server sizes, at each of the user densities, for both Knowledge Worker and Task Worker workloads.

Small D2v2 Instance Results: Knowledge Worker

In the following charts, we see the inflection point at about 20 user sessions, which also keeps the average application open time below 4 seconds. CPU and IOPS, however, remain well within acceptable limits.



D2v2 Instance, Knowledge Worker
Average Application Open Time (Sec)

D2v2 Instance, Knowledge Worker
Average Steady State and Peak CPU Utilization

D2v2 Instance, Knowledge Worker
Steady State Total Disk Read/Write Operations Per Second

Figure 4: Test Results for D2v2 Knowledge Worker

Small D2v2 Instance Results: Task Worker

In the following charts, we see the inflection point for CPU and IOPS at about 26 sessions. The app open time remains fairly constant, and is well below the 4-second limit.

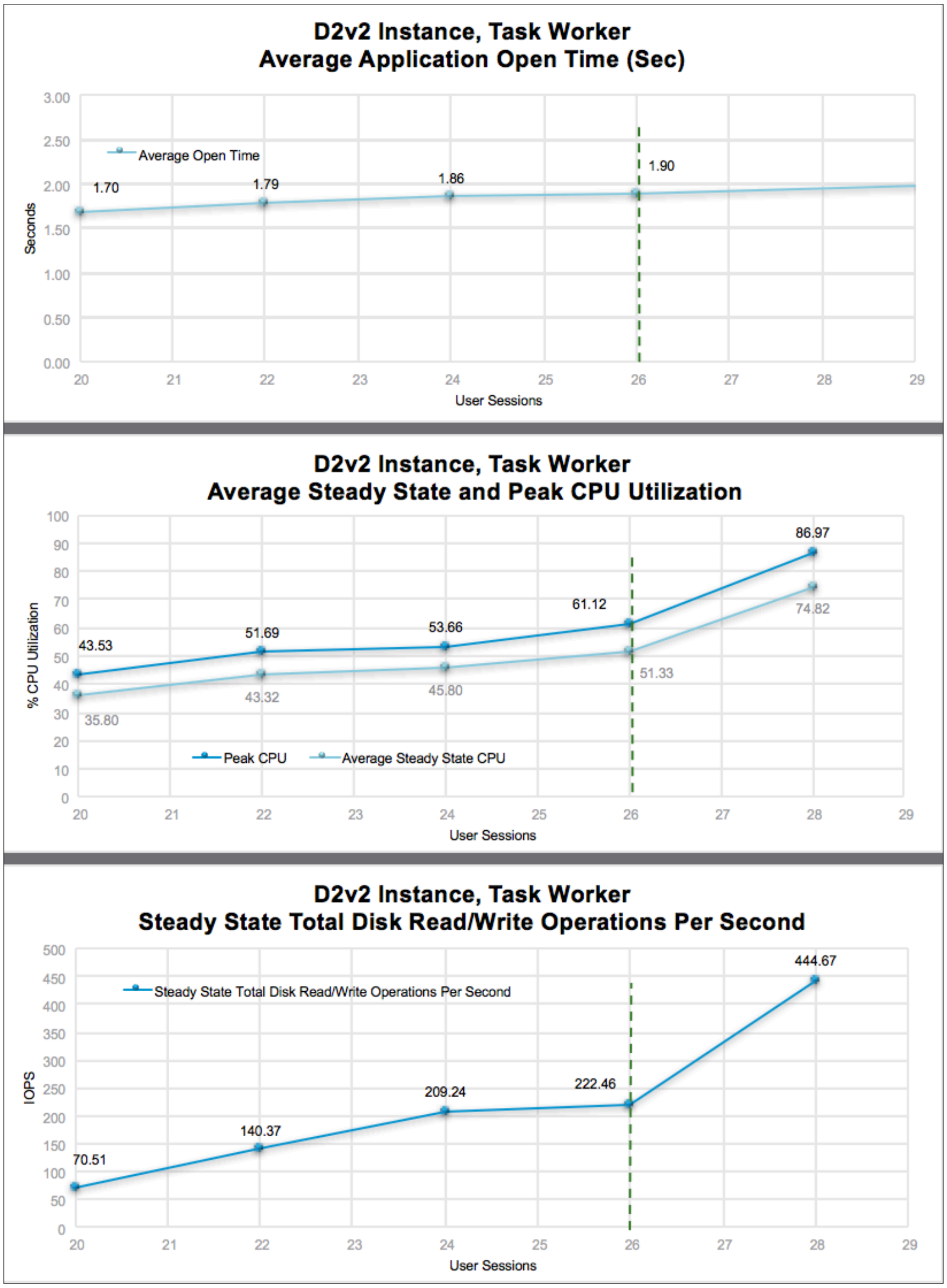


Figure 5: Test Results for D2v2 Task Worker

Medium D3v2 Instance Results: Knowledge Worker

In the following charts, we see the inflection point at about 30 user sessions, which also keeps the average application open time below 4 seconds. In addition, above this user density the CPU Utilization and Average Steady State starts to plateau, which indicates saturation and likely degraded user experience.

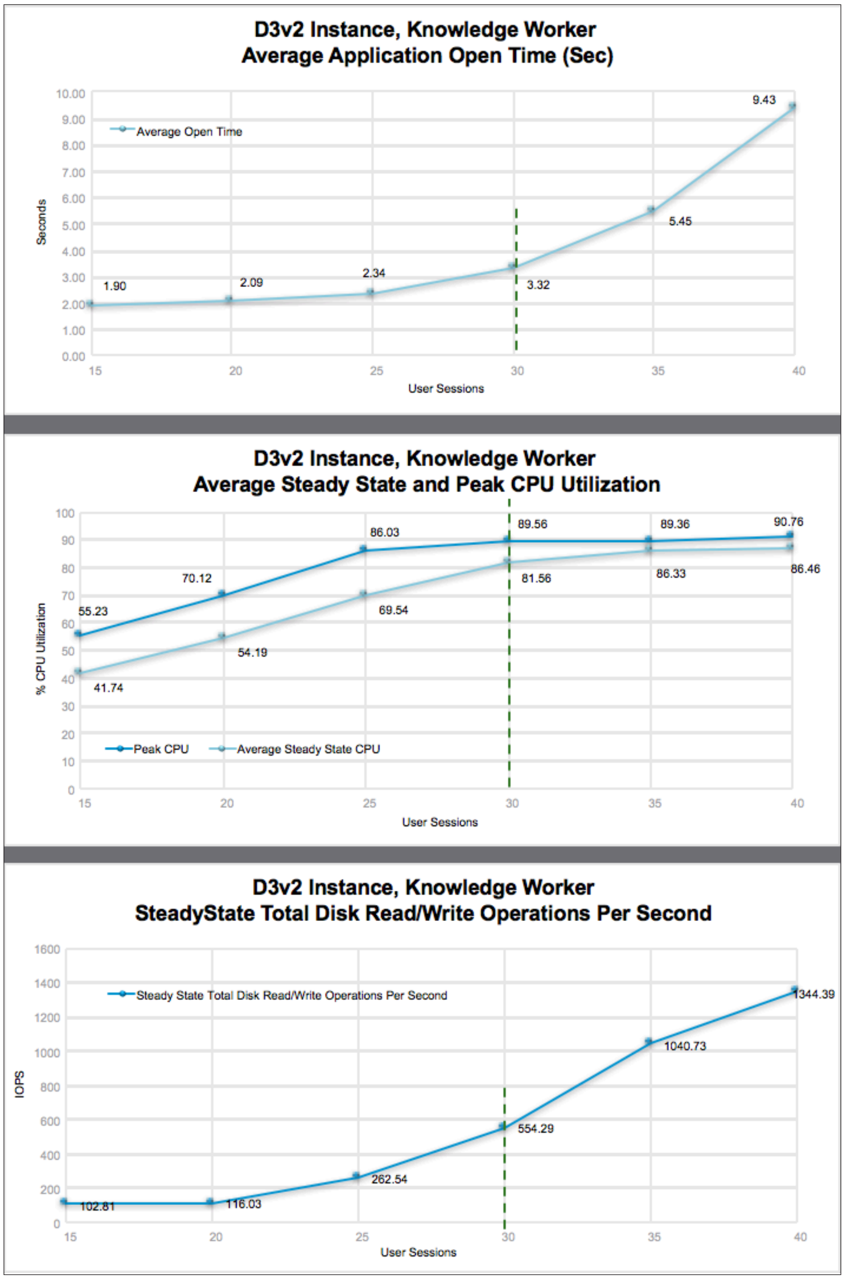


Figure 6: Test Results for D3v2 Knowledge Worker

Medium D3v2 Instance Results: Task Worker

Similar results to that of the D3v2 Knowledge Worker workload were obtained for the D3v2 Task Worker scenario, however the inflection point was found to be at around 50 sessions. Application launch time was well below the 4-second limit.

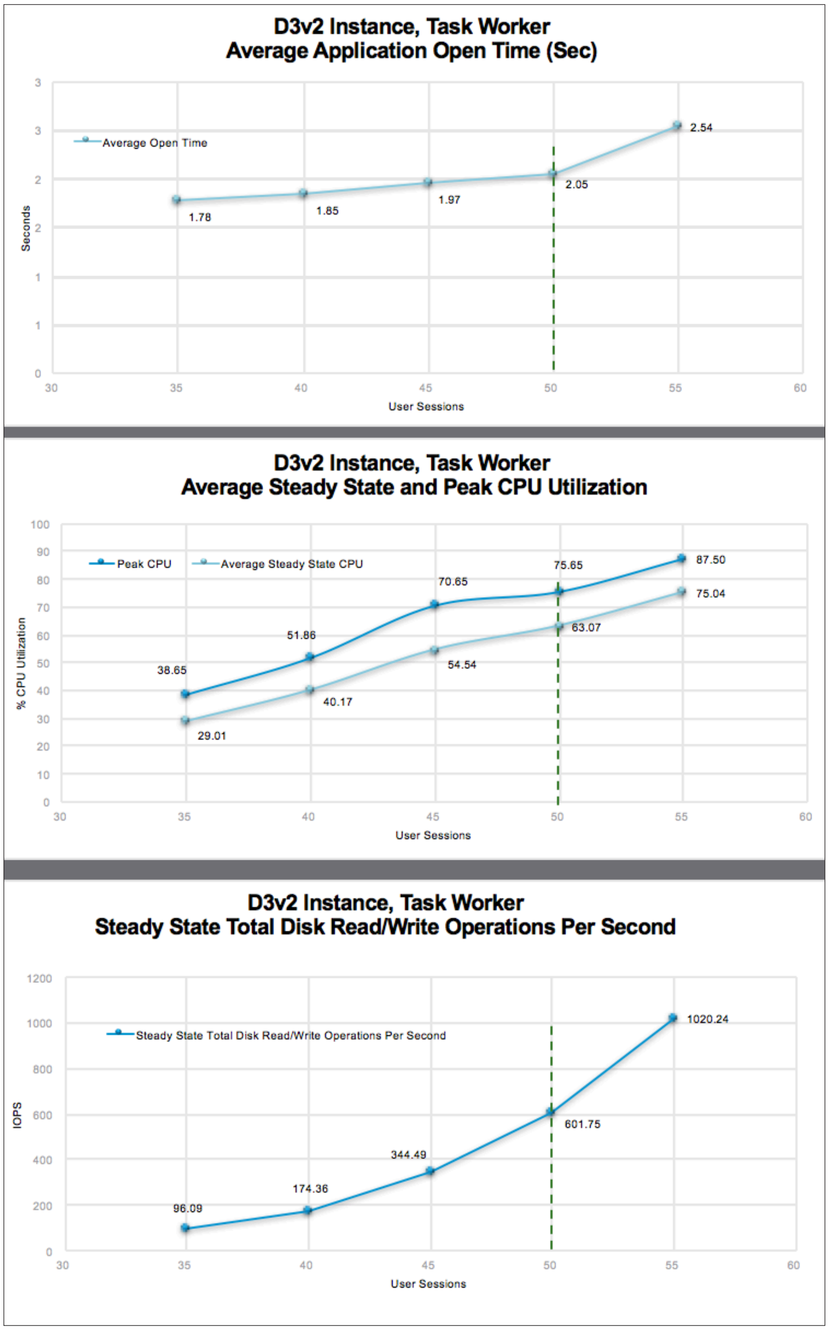


Figure 7: Test Results for D3v2 Task Worker

Large D4v2 Instance Results: Knowledge Worker

The inflection point for IOPS seems to be between 60 and 65 sessions, and at 70 sessions you see the CPU saturating, indicating poorer user experience. At 65 sessions, we are still below the 4-second application open-time threshold. However, the engineering team felt that recommending 60 sessions for this size would be preferable to maximize user experience.

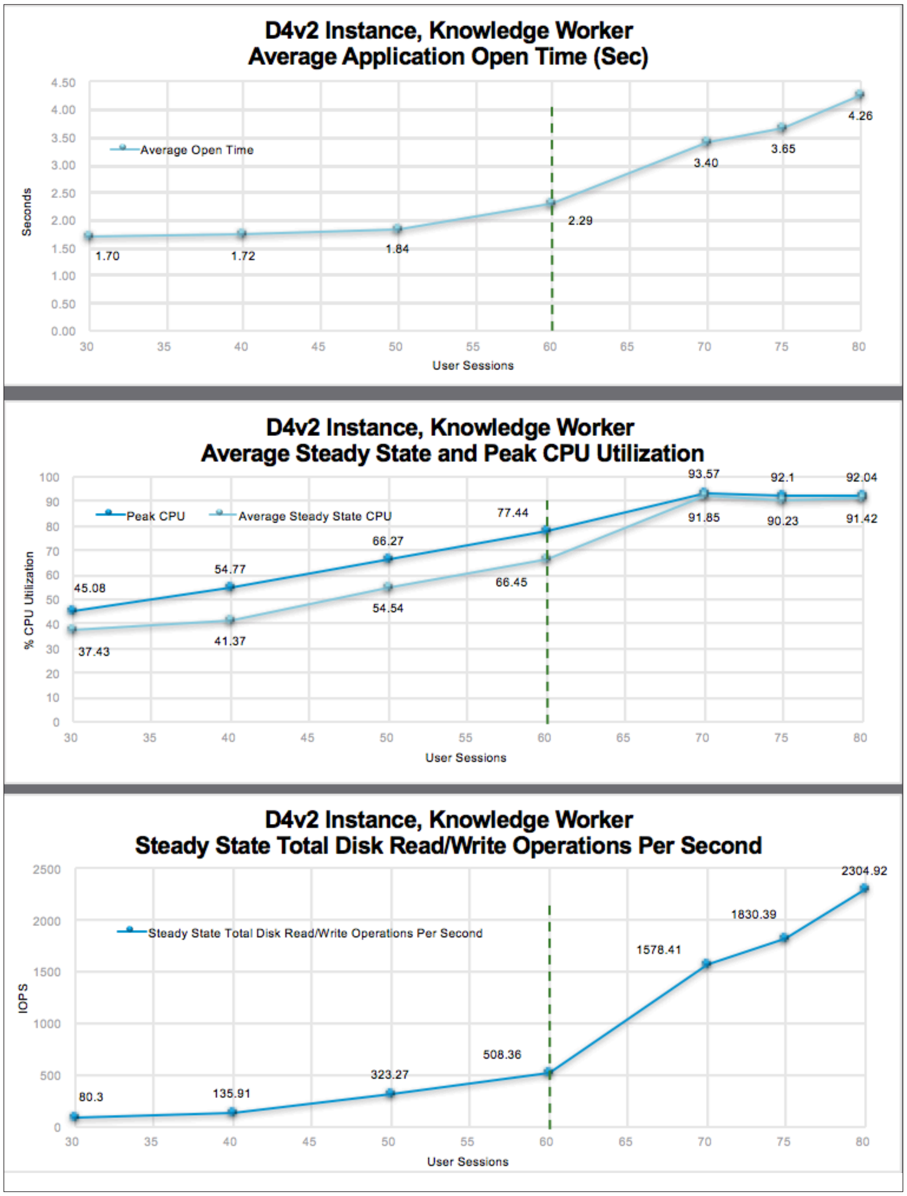


Figure 8: Test Results for D4v2 Knowledge Worker

Large D4v2 Instance Results: Task Worker

This was probably the hardest test to run and to reach a suitable conclusion. At all densities tested, the application open times were acceptable, and CPU and IOPS looked good. There is the start of an inflection point for CPU at 85 sessions, and as a result, our recommendation here is 85 sessions.

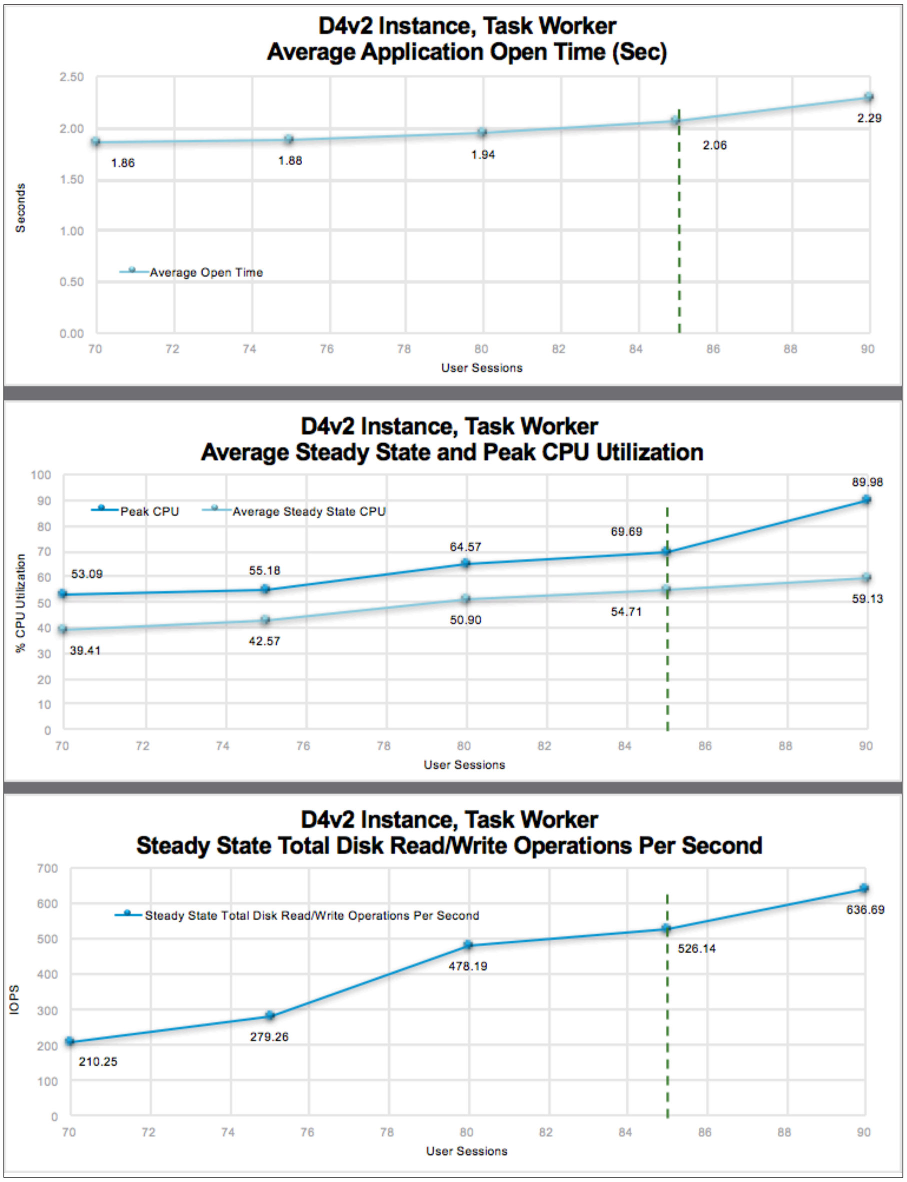


Figure 9: Test Results for D4v2 Task Worker

GPU NV6 Instance Results: Knowledge Worker

Due to the driver limitations with the NV6 card, it is limited to a maximum of 20 sessions on Windows Server 2012 R2. When VMware Horizon 7 supports Windows Server 2016 with GPU-backed cards, we will retest and provide additional numbers. As would be expected, the NV6 server handles 20 sessions with ease, and is by no means really limited to 20 sessions, other than due to the driver limitation.

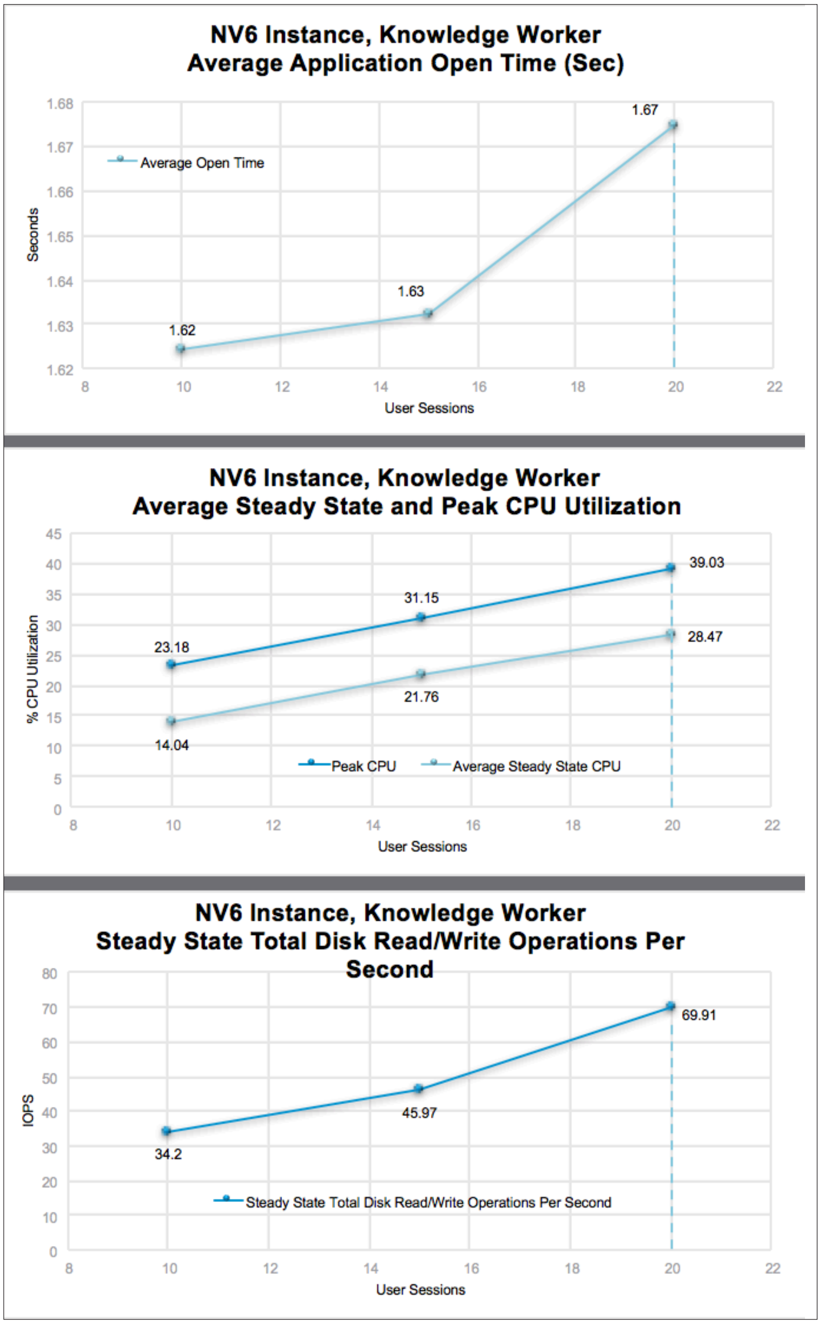


Figure 10: Test Results for NV6 Knowledge Worker

Power Management and Cost Analysis

Enterprises typically don't want to serve applications or desktops to only 60 users, and as such they will need more than one server in a farm to meet the user demand. One of the considerations, then, is how many servers are needed to meet that demand.

Let's take an example of an enterprise wanting to deliver applications to 1,000 knowledge workers where they have determined that they can choose Large servers with 60 sessions per server. This would require 1000 ÷ 60 servers or ~ 17 servers. This assumes, though, that they need those 1,000 sessions to be active 24x7. This is usually not the case, and as such, they could potentially have many fewer servers knowing that they will be unlikely to need to deliver all 1,000 sessions concurrently, all of the time. This is the concept of under-provisioning: for example, they may choose to only stand up 10 servers, offering a maximum of 600 concurrent sessions, but with the knowledge that over a period of 24 hours, they may serve a full 1,000 sessions. With on-premises delivery of RDS applications and desktops, due to the cost of purchasing expensive servers, this under-provisioning method is an approach often taken by enterprises. If more than 600 users tried to log in concurrently, then those additional users would be unable to get access to the capacity.

With cloud delivery of infrastructure, there is no longer the up-front cost of buying expensive servers. Capacity is available on demand. Horizon Cloud Service on Microsoft Azure will take advantage of this while allowing costs to be controlled.

If a RDS Server has even just one (1) active session, then it cannot be powered off, and it is not possible to migrate sessions between RDS servers. So, in the preceding example, if the enterprise had 17 servers, each with a single active session, then none of those servers could be powered off.

In Microsoft Azure, a VM that is powered off (and deallocated) does not incur compute charges. Even though such a VM still incurs storage costs, the cost of compute is far greater than storage. As a result, if the farm's servers can be powered off and deallocated, that will reduce the monthly bill significantly.

Horizon Cloud Power Management

Horizon Cloud provides power management capabilities for the Microsoft Azure servers. Farms can be configured to have a minimum number of servers, and a maximum number of servers, along with a sessions-per-server value.

The maximum number of servers multiplied by the sessions per server determines the maximum user population that can be served. The minimum number of servers determines the minimum servers that will be powered on.

For example:

```
Consider;  Min = 1, Max = 10, #Sessions per Server = 20
This configuration will allow a maximum of
Max #Sessions == 10 * 20 == 200 sessions if all the servers are powered on. But if 19
servers are powered off, then there is still 1 server powered on allowing up to 20 users
to connect. When 19 servers are powered off, the cost will be just for 1x Server compute
charges + 20 server storage charges, plus any network ingress/egress charges.
(Significantly less than paying for 20 server compute charges + 20 storage charges!)
```

Horizon Cloud also allows the minimum number of servers to be set to 0. This means that when no users require the service, then all servers will be powered down automatically. Note that this can result in a poorer experience for the next handful of users to log in, since they will need to wait for the server to power on, boot, and reach a state ready to receive incoming connections (usually around 5 minutes).

A freshly deployed server will take approximately 15 minutes to reach a steady state on first boot. That behavior is because the first ever boot, plus domain join (and associated restarts), takes some time to complete. In order to minimize the impact on users logging in, with servers powering on, Horizon Cloud will pre-provision the `maximum` number of servers in a farm upfront, allow them to do their first boot, domain join, and more, and then power them down. This method has two benefits:

1. The farm is created at the maximum size. Due to the way that the quotas and limits work in Microsoft Azure, these resources (such as CPUs) are reserved for use, and this guarantees that the servers can be powered on when required. As such, the maximum user load for a farm can always be met.
2. By pre-provisioning, that initial first boot time of 15 minutes is performed ahead of the need, meaning that subsequent power cycles are much faster (around 5 minutes).

Utilization of a farm is calculated by taking the number of active sessions within a farm and dividing by the total number of sessions possible from the powered-on servers in that farm. For example, if there is 1 server powered on, with a maximum of 10 servers with `#Sessions per server=20`, and `Number active sessions=10`, then the utilization is 10 ÷ (20*1) = 50%. If, however, 2 servers were powered on, then the Utilization = 10 ÷ (20 * 2) = 25%.

Note: When looking at the node level, the occupancy is calculated without regard for the number of servers powered off. That is, it looks at the number of active sessions and the maximum possible number of sessions given the farm configuration that has been made, to calculate a percentage. In the preceding example with 10 sessions, 20 sessions per server, and a maximum of 10 servers, the node level utilization would be calculated as 10 ÷ (20*10) = 10 ÷ 200 = 0.5%.

The Horizon Cloud power management algorithms offer three modes of operation:

- Optimized Performance
  - Low threshold: 27%
  - High threshold: 50%
- Balanced
  - Low threshold: 37%
  - High threshold: 66%
- Optimized Power
  - Low threshold: 55%
  - High threshold: 80%

These settings adjust the point at which servers will be powered on (and powered off) to benefit the overall cost, provide a better user experience, or find a balance between the two.

Specifically, they adjust the power-on threshold values using the high threshold, such that once the farm utilization exceeds that threshold, a new server will be powered on. The high threshold value is used in both the power-management algorithm as well as the session-placement algorithm.

When a new session request comes, we place the session on the most loaded server that is below the high-threshold percentage. If all servers hit the high threshold percentage value, then we place the session in a round-robin manner: we choose the server that is above high threshold and has fewest number of sessions.

For example, if the high threshold was set at 50%, in a farm with 20 sessions per server, then once the eleventh user connects, the threshold is exceeded (since occupancy >50%) and a new server will be powered on.

Optimized performance will power on a new server much sooner; this tries to ensure that as more users log in, no one will need to wait for the server to complete powering up.

Optimized power will power on a new server much later, when the occupancy hits a much bigger number, thus leaving more servers powered down for longer. However, this choice creates the possibility that new users connecting might need to wait for the server to power on before they can connect.

Balanced mode tries to take the middle ground of finding good performance and good power management. However, in all three cases, the success of these approaches will vary based on what sort of rate of user logins you expect during the day. If you have a login storm at 9AM, then it is likely that setting enough minimum servers will be needed to ensure that there is enough capacity to meet that immediate demand.

In order to properly determine the required number of servers, it is important to consider the typical login/logout rate of users in your enterprise. Consider the two scenarios that follow: User Profile A and User Profile B.

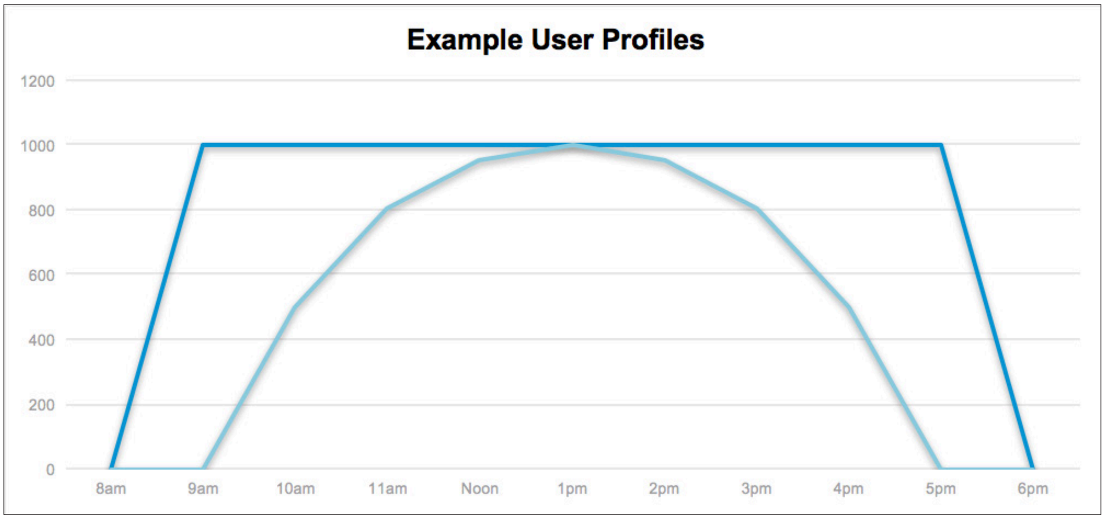


Figure 11: Two Example Enterprise User Profiles for Login/Session Concurrency

In User Login Profile A, all users log in at 9AM, and all log out at 6PM. This user distribution essentially requires a constant number of servers to be powered on during office hours, and requires no servers at night. Optimized performance will be the best mode for such user behavior; however, it is worth noting that Horizon Cloud has some exciting new power-management modes coming in a future release to even better handle this user distribution. User Login Profile B has a much smoother login/logout rate, and this rate will much better fit the Optimized Power or Balanced mode of operation. Because during the morning and late afternoon, a good number of servers could power off, thus saving costs, with only the maximum number of servers being powered on around midday.

It is worth noting that powering servers off is an interesting challenge. Because sessions cannot be moved between servers, it means that a server can only be powered off once all sessions have been disconnected. For some enterprises, forcibly disconnecting users at 6PM is acceptable, for example, a call center where shifts have finished, and there is nothing stateful (such as unsaved documents) that could be lost from a forced logout. For this use case, Horizon Cloud provides granular timeout controls to enable such timely logouts to be done. For other use cases, where knowledge workers may be producing documents and more, forced logouts are most likely unacceptable. Timeouts can still be set—for example, after 8 or 24 hours of inactivity—and users can be educated that they will be logged out after such extended periods of time.

The power-management features of Horizon Cloud will do their best to allow as many servers to power down during the day as possible. Once Horizon Cloud detects that the user population is shrinking, and starting to drop below the selected threshold, then it will target some of the least loaded servers for quiescing. One or more servers are placed in the quiesced state, meaning that any active sessions on those servers continue uninterrupted. The users do not know this is happening. Once the server is in quiesced mode, no new sessions will be placed on this server. This means that once those few users eventually disconnect, that server will be powered off and deallocated, thus saving costs.

As an enterprise, it is therefore important to understand the typical user behavior of your users over the course of a working week. This will help proper sizing to ensure optimal user performance, while minimizing costs. The Utilization Report feature of Horizon Cloud might be useful here. Specifically, a farm could be created for a few weeks with the maximum value being set equal to the minimum value (that is, with power management disabled). Then, the Utilization Report will show the typical usage patterns over the week, and from that it is possible to work out what the minimum number of servers should be specified at for a given farm.

Microsoft Azure Costs and Sizing Considerations

Pricing for Microsoft Azure virtual machines varies by region and also varies by size and family of VMs. The pricing can include licensing, however with Microsoft Windows Azure Hybrid Use Benefit licensing it is possible to port licenses from on premises to the cloud. Details of the Microsoft Azure pricing (which regularly changes) is available on the Microsoft website.

While some numbers are presented here, they should be considered for illustration only, because cloud prices change frequently, and vary by region and may be affected by your specific agreement with Microsoft Azure.

Upon initial thought, it might seem most sensible to pick the largest servers to serve 1,000 users, because this will require fewer servers to serve up the capacity. Certainly, when running on premises, this likely makes the most economical sense because it requires less rack space and less networking. However, with RDS, remember that a server can only be powered off when there are no user sessions running on the server. As such, a server that normally has 80 sessions running on it will most likely take longer for those sessions to finish than a server with just 20 sessions. That is, in Microsoft Azure it is likely that having a greater number of smaller servers gives the greatest chance that servers can be powered down, and therefore will save money. This section explores this concept, and includes some approximate calculations to prove it.

The costs of compute for the Dv2 Family of VMs at the time of this writing are summarized in Table 4.

VM SIZE	CPUS	MEMORY	DISK	PRICE / HOUR
D1 v2	1	3.50 GB	50 GB	$0.14
D2 v2	2	7.00 GB	100 GB	$0.28
D3 v2	4	14.00 GB	200 GB	$0.56
D4 v2	8	28.00 GB	400 GB	$1.12
D5 v2	16	56.00 GB	800 GB	$2.02

Table 4: Dv2 Family Compute Costs

Interestingly, for this Dv2 Family of VMs, if you chart the cost/CPU core, it is roughly linear, as shown in Figure 12. This would indicate that you might not need to worry about which server size is selected for price modelling.

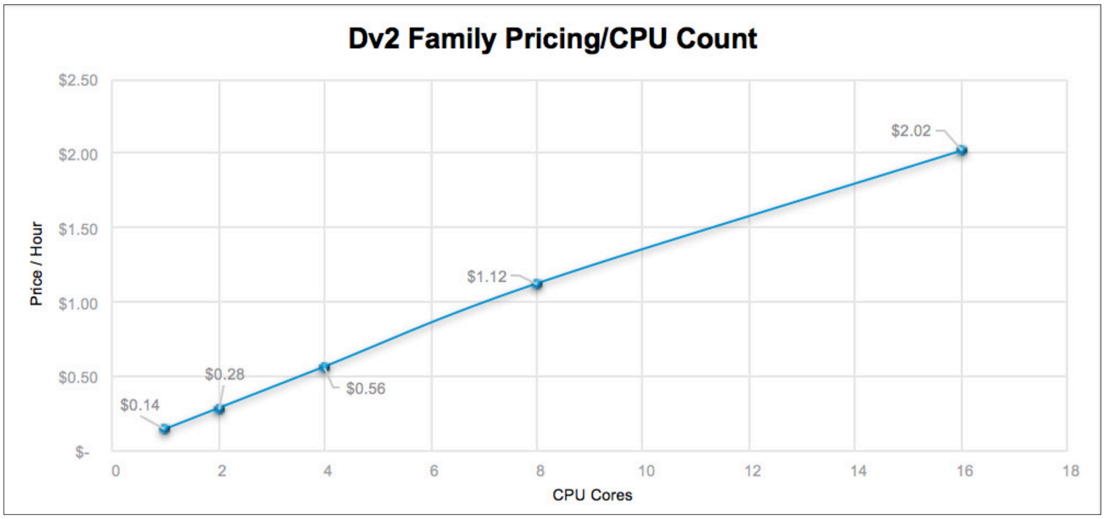


Figure 12: Dv2 Family – Cost per CPU Core per Hour

However, even though the cost per CPU core is almost linear, it is not necessarily the case that the cost per user is also linear. From the preceding testing results, the number of users per server did not linearly scale per CPU (because the memory did not exactly scale linearly). As such, Figure 13 shows the cost per hour per user for each of the VM sizes when applying the user density results as identified in the tests.

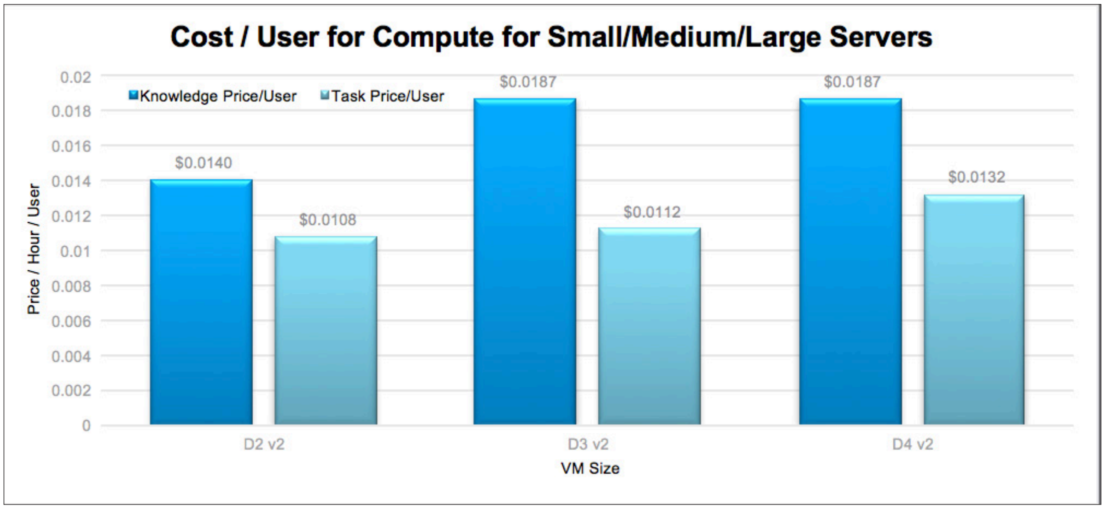


Figure 13: Dv2 Family – Cost per User per Core per Worker Type

Figure 13 clearly shows that the lowest cost per user is achieved from the smallest server D2v2. However, based on the discussion in the Power Management section, this should be considered carefully, because with a farm configured for 20 knowledge workers per server, it may mean that users have to wait more often for new servers to power on, because it does not take many users to log in before a new server is required to handle additional load.

Let's assume, however, for the purposes of cost analysis, that the login rate or user experience is such that using small servers is equally acceptable as large servers.

Example scenario: 1,000 knowledge workers. Should the enterprise choose many smaller D2v2 servers or fewer larger D3v2 or D4v2 sized servers? If the use case was 1,000 task workers, would that change the selection of server size?

For Knowledge Workers, let's use the results from this white paper; specifically, let's start by calculating the number of servers required to serve this load:

Small D2v2: 20 knowledge worker sessions/server → 1000 ÷ 20 = 50 servers

Medium D3v2: 30 knowledge worker sessions/server → 1000 ÷ 30 = 34 servers

Large D4v2: 60 knowledge worker session/server → 1000 ÷ 60 = 17 servers

There is an hourly cost for the VMs to be powered on, plus there is a storage cost for the OS disk image. For the Dv2 family, this is at a Standard storage tier, and equates to $5.90/month per VM (in East U.S. region). This storage cost applies even if the VM is powered off.

Let's now assume that we have a workload profile similar to that of the fictitious User Profile A as shown in Figure 11; that is, everyone logs in together and logs out together. Let's also assume the users work only a 40-hour week.

Table 5 shows the overall price for this fictitious example covering compute and storage. (Data ingress/egress charges apply on top of this, along with the base infrastructure cost of the Horizon Cloud environment.)

KNOWLEDGE WORKER	SESSIONS / SERVER	NUMBER OF SERVERS REQUIRED FOR 1,000 SESSIONS	COMPUTE COST / HOUR	128 GB HDD DISK COST / SERVER / HOUR ($5.90/ MONTH)	PRICE / HOUR FOR 1,000 USERS OF COMPUTE + DISK	PRICE / MONTH (100% POWERED ON)	PRICE / MONTH*	PRICE / USER / MONTH*
Small D2v2	20	50	$0.28	$0.01	$14.41	$10,520.90	$2,727.57	$2.73
Medium D3v2	30	34	$0.56	$0.01	$19.32	$14,105.70	$3,506.77	$3.51
Large D4v2	60	17	$1.12	$0.01	$19.19	$14,005.40	$3,406.47	$3.41
*Assumes 40-hour week.								

Table 5: Example Costs for 1,000 Users with Different VM Sizes Using Knowledge Worker

Even without any power management, the table above clearly shows that running more smaller servers is the most cost-effective option based on the user densities configured. Once power management is enabled, then the smaller server scenario remains the most cost effective.

Table 5 clearly shows that without power management, the running costs of enough VMs to serve 1,000 users for just 40 hours of work per week would be around 400% more than they need to be if power management can be applied.

If we repeat these calculations for the task-based worker user densities, then the calculations look like this:

TASK WORKER	SESSIONS / SERVER	NUMBER OF SERVERS REQUIRED FOR 1,000 SESSIONS	COMPUTE COST / HOUR	128 GB HDD DISK COST / SERVER / HOUR ($5.90/ MONTH)	PRICE / HOUR FOR 1,000 USERS OF COMPUTE + DISK	PRICE / MONTH (100% POWERED ON)	PRICE / MONTH*	PRICE / USER / MONTH*
Small D2v2	26	39	$0.28	$0.01	$11.24	$8,207.60	$2,128.80	$2.13
Medium D3v2	50	20	$0.56	$0.01	$11.37	$8,299.90	$2,065.23	$2.07
Large D4v2	85	12	$1.12	$0.01	$13.55	$9,887.90	$2,406.30	$2.41
*Assumes 40-hour week.								

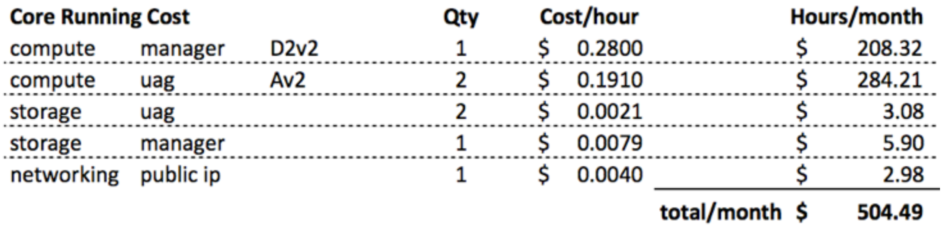
Table 6: Example Costs for 1,000 Users with Different VM Sizes Using Task Worker

Interestingly, due to the user density values for the Medium server, for a task-based worker the calculations show the Medium server would provide the most economical size by a small margin. Given the cost delta is very small in these calculations, it may be advantageous to just use the small servers across all farms, rather than using a mix, because that might make ongoing management and maintenance a bit easier. Similar calculations can be done for a smaller number of users too, which will give similar (but potentially different) results!

If the user profile for login rates was a more gradual ramp up in the morning and ramp down in the evening, then even greater savings could be had, because there would be no need to have all servers powered on at the start of the day.

Clearly, by using small servers in the knowledge-based worker example, even though you require more of them, the cost benefits from power management mean that small servers are much more economical. The key consideration here is whether having just a small server gives sufficient head room for the required login rate of the user population, without undue periods waiting for servers to power on.

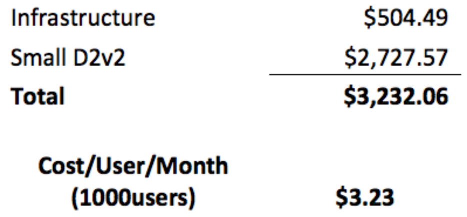
The following calculations show the approximate cost for the base appliances that are required in Horizon Cloud:



Core Running Cost			Qty	Cost/hour	Hours/month
compute	manager	D2v2	1	$ 0.2800	$ 208.32
compute	uag	Av2	2	$ 0.1910	$ 284.21
storage	uag		2	$ 0.0021	$ 3.08
storage	manager		1	$ 0.0079	$ 5.90
networking	public ip		1	$ 0.0040	$ 2.98
				total/month	**$ 504.49**

Figure 14: Example Running Costs for Always-On Infrastructure

Bringing this together, assuming the Small D2v2 server provides an acceptable user experience for delivering the 1,000 concurrent sessions, then the monthly cost per user calculates to:



Infrastructure	$504.49
Small D2v2	$2,727.57
Total	**$3,232.06**
Cost/User/Month (1000users)	**$3.23**

Figure 15: Example Costs for 1,000 Users Including Infrastructure

It is important to note however that Microsoft Azure does have some additional costs that are hard to quantify here in this example. For example, network ingress/egress costs (which would include delivering the remote session to the users using BEAT, Blast, or PCoIP), along with additional storage or IO charges, and any and additional software licensing costs.

Note: The previous costs and user densities serve as an example only, and individual results, costs per region, and scenarios may vary.

Conclusion

This white paper introduced the VMware Horizon Cloud Service on Microsoft Azure platform. It then explained how user density testing was performed for RDS servers running in Horizon Cloud, and detailed tests were performed to identify the recommended user densities for both Knowledge and Task Worker workloads on various VM sizes in Microsoft Azure. We then presented how power management works within Horizon Cloud, which can produce great savings by powering down servers when they are not needed. Finally, an example cost exercise was performed for a fictitious use case of 1,000 users. The exercise helped to identify some of the considerations, and demonstrated that using more smaller servers is often more economical than using fewer large servers.

Authors

Peter Brown has been with VMware since 2012. Peter is a Director of R&D in the End-User Computing Business unit, and is currently the engineering lead for the Horizon Cloud Service with Microsoft Azure. Peter has worked with Horizon 7 (on premises) and Horizon Cloud Service, and has led engineering efforts for innovations such as True SSO, Linux Desktops, USB Redirection, RTAV, Serial and Scanner Redirection, and much more.

Fred Schimscheimer has been with VMware since 2007. Fred is a staff engineer in the VMware EUC Office of the CTO. Fred helps prototype and validate advanced development projects and is an expert in storage and workloads for virtual desktop solutions. Fred is the author of RAWC, VMware's first desktop Reference Architecture Workload Simulator.

Contributor

Keerthi Singri has been with VMware for 13 years, advancing from an intern to Sr. R&D Manager, and has worked on custom portals, web applications, VMware ThinApp Factory, and VMware Blast HTML Access. In the last few years, he has managed teams for Application remoting, Windows installers, and Horizon Cloud, with a recent focus on Horizon Cloud on Microsoft Azure.

VMware, Inc. 3401 Hillview Avenue Palo Alto CA 94304 USA Tel 877-486-9273 Fax 650-427-5001 www.vmware.com