ORACLE ON AZURE IAAS RECOMMENDED PRACTICES FOR SUCCESS

Kellyn Gorman, Oracle SME on Azure IaaS
Cloud Architecture and Engineering Data/Infra, Microsoft

Contents

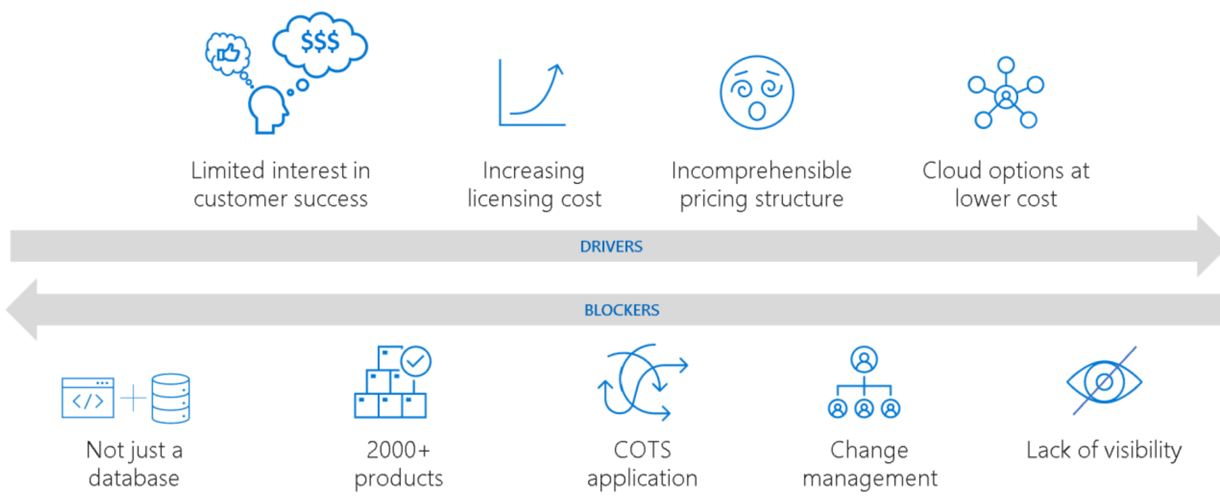
Why Oracle on Azure

What is Oracle - More than just a database

Apps, database, hardware, virtualization, and cloud

Oracle can often present hurdles to the cloud due to the complexity, size and high IO demands of the workload. Although this paper focuses on the database, please understand that Oracle can present itself as a multi-tier system, web code, applications, hardware, etc.

Throughout this paper, Oracle databases will continually be discussed as "workloads". We have discovered that by focusing on the Oracle workload and not on the database we are much more successful. Azure presents an opportunity to migrate Oracle workloads to match on-premises in an Infrastructure as a Service, (IaaS) model, but to do more with this critical data regardless of if it's a future of analytics, data lake, global data governance, machine learning or even artificial intelligence.



Oracle, with all its moving parts, can be an overwhelming project to start planning to migrate to the cloud. The goal of this paper is to break down each piece around the database workload tier and give a meaningful starting point and steps to achieving what we and our clients must accomplish.

Lift and Shift the Workload

Over-provisioned

Oracle does not appear to make it easy to migrate anywhere but Oracle Cloud, (OCI):

[Penalizing hypervisor virtualized CPUs](#).

*"**Microsoft Azure** – count two vCPUs as equivalent to one Oracle Processor license if multithreading of processor cores is enabled, and one vCPU as equivalent to one Oracle Processor license if multi-threading of processor cores is not enabled."*

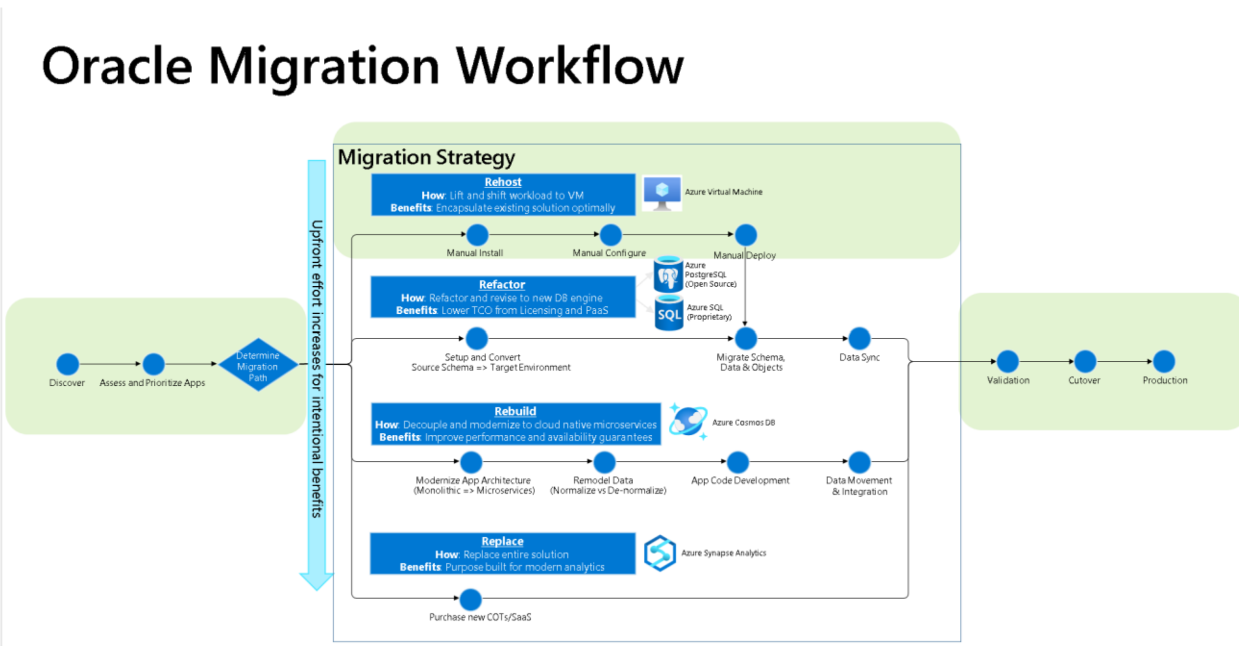
The reason for this 2:1 penalty should not faze a customer coming to the Azure cloud. We have proven how on-premises database hosts are sized out for capacity planning and there's a definitive pattern. It is common (for multiple reasons) for these hosts to be considerably over-provisioned vs. what they require to run the workloads. Most often due this is due to:

1. How on-premises hardware must be sized/padded to meet resource needs for years vs. the ability to scale on CPU, like the cloud. This results in requirements for on-premises hosts to be larger than required, at the time of purchase.
2. DBAs are instructed to size out the on-premises hardware to support the database for 2-7 years and must use both capacity growth values and assumptions to estimate what those resource needs will look like.
3. Knowing how budgets work, DBAs also expect that there is a considerable chance when the database comes up for a hardware refresh, it will not receive the funds in the budget, forcing the team to run for longer on the original hardware.   As such, DBAs tend to pad the original numbers to prepare for this.
4. Workloads have changed and in recent years, transactional systems have morphed into hybrid environments with higher IO workloads and better CPUs have offered us better performance with less demand for upgrades.

Considering the above list, Azure customers have proven around 85% of Oracle workloads assessed will require a fraction of the vCPU allocated to the on-premises systems. The [Automatic Workload Repository](#) (AWR) is very good at identifying a [solid workload and with a worksheet](#) that can adjust for averages and aggregate values, will provide clear estimates to size out the workload for the Azure cloud.

Oracle on IaaS or an Azure Native PaaS Solution

There are numerous migration paths that Oracle workloads can take in your Azure journey, but the one we're going to focus on is Oracle is staying on Oracle, (even if not permanently) in the Azure cloud.



Before Oracle workloads can be considered for refactoring to a native PaaS solution, an assessment should be performed by the Azure Data Migration Assessment tool and reviewed with the proper specialists, but for this paper, the focus is for those workloads that are running Oracle on Azure only.

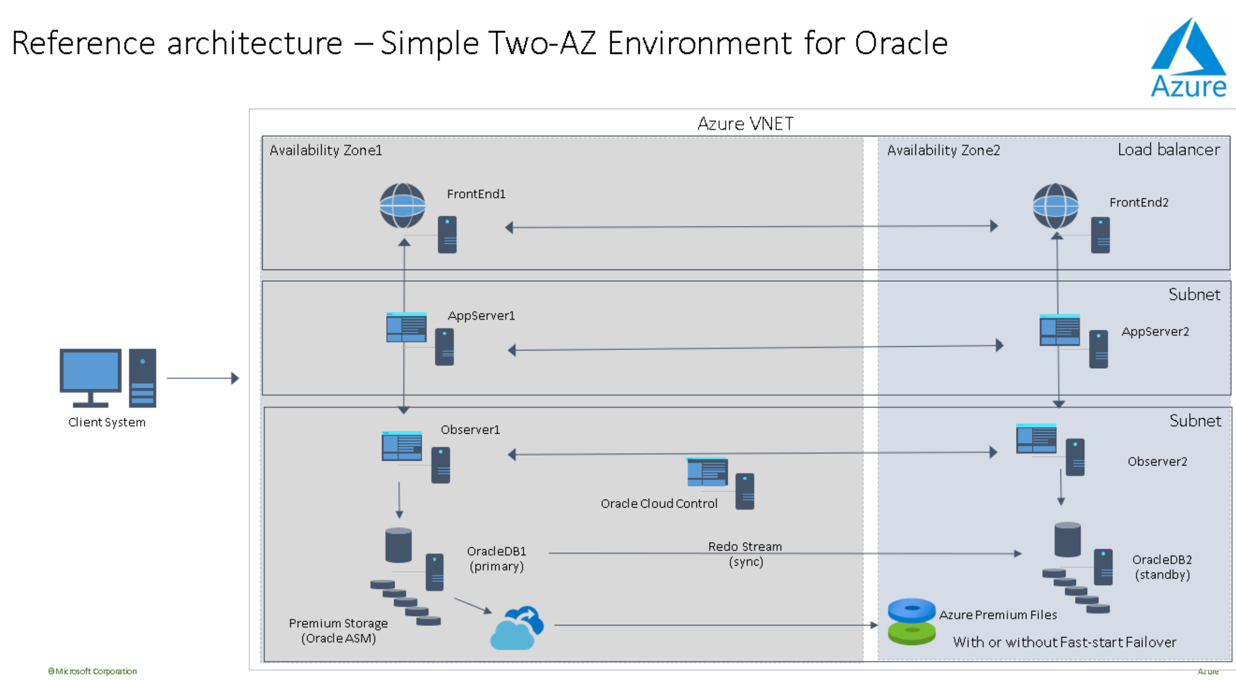
Sizing for the Application or Middleware Tier

At a high level, we do need to acknowledge Oracle applications are made up of multiple services, which can be hosted on the same or multiple virtual machines in Azure. When an Oracle workload is brought onto Azure, the application tier is commonly an infrastructure motion, sizing out similarly to the on-premises environment, then leveraging the Oracle on Azure specialist to help with the "heavier" migration effort for the database tier in conjunction with the adjacent teams.

Although Oracle is significantly less demanding with application and middleware tiers, they place great value at the database per core licensing, requiring more focus on rightsizing when migrating the workload to the cloud. Even for those application tiers that are running on large, engineered Exalogic systems, with a virtualized, Oracle Virtual Manager, (OVM) layer, it's quite simple to translate directly to VMs in Azure. The database layer is much more challenging.

Oracle Application instances can be set up using best practices for IaaS workloads in Azure, including the use of private or public endpoints once migrating to Azure for connectivity. Both Microsoft and Oracle recommend setting up a *bastion host VM* with a public IP address in a separate subnet for management of the application.

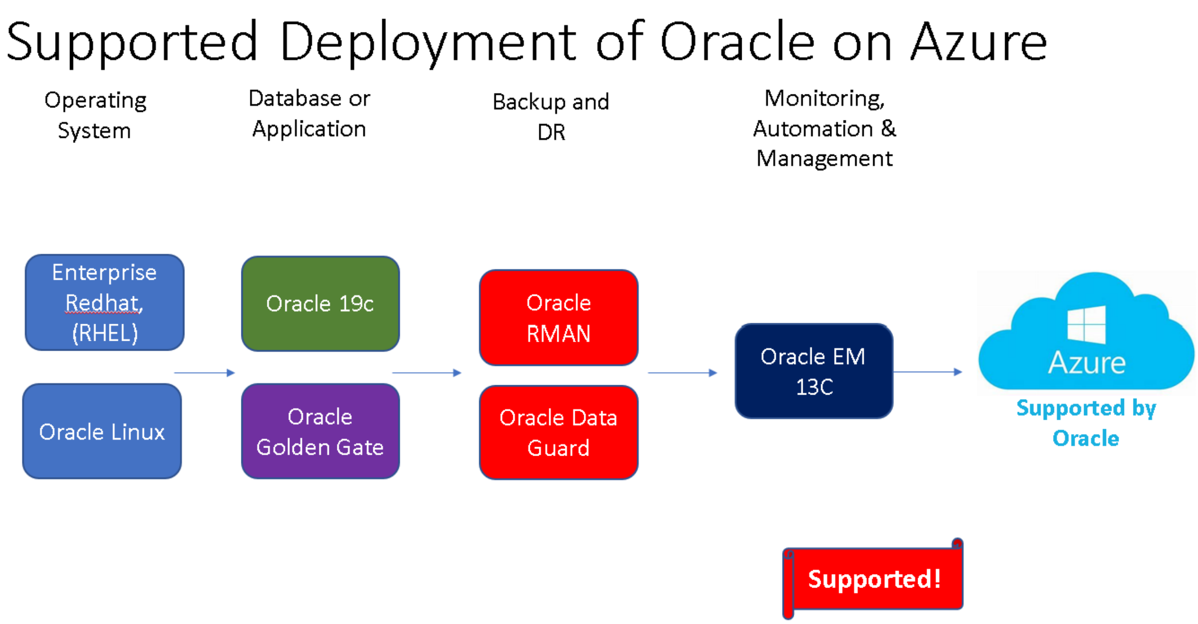
For added security, set up network security groups at a subnet level to ensure only traffic on specific ports and IP addresses is permitted. For example, machines in the middle tier should only receive traffic from within the virtual network. No external traffic should reach the middle tier machines directly.



For high availability, you can set up redundant instances of the different servers in the same availability set or different availability zones. Availability zones allow you to achieve a 99.99% uptime SLA, while availability sets allow you to achieve a 99.995% uptime for the database tier in-region.

Operation System Choices

For Operating Systems, such as Oracle Linux, there isn't any licensing costs, only support costs to continue using it in Azure, so if the customer isn't locked into Azure Monitor for these Linux VMs the Oracle workloads are running on, Oracle Linux or RedHat, (with additional licensing cost through RedHat) are the recommended options.



Although Windows, SLES and other Unix platforms are supported operating systems for Oracle workloads to run on, it is highly recommended to use a supported version of Oracle Linux or RedHat, which receives the most support from Oracle.

Bastion tier

The bastion host is an optional component that you can use as a jump server to access the application and database instances. The bastion host VM can have a public IP address assigned to it, although the recommendation is to set up an ExpressRoute connection or site-to-site VPN with your on-premises network for secure access. Additionally, only SSH (port 22, Linux) or RDP (port 3389, Windows Server) should be opened for incoming traffic. For high availability, deploy a bastion host in two availability zones or in a single availability set.

You may also enable SSH agent forwarding on your VMs, which allows you to access other VMs in the virtual network by forwarding the credentials from your bastion host. Or use SSH tunneling to access other instances.

Here's an example of agent forwarding:

ssh -A -t user@BASTION_SERVER_IP ssh -A root@TARGET_SERVER_IP`

This command connects to the bastion and then immediately runs ssh again, so you get a terminal on the target instance. You may need to specify a user other than root on the target instance if your cluster is configured differently. The -A argument forwards the agent connection so your private key on your local machine is used automatically. Note that agent forwarding is a chain, so the second ssh command also includes -A so that any subsequent SSH connections initiated from the target instance also use your local private key.

Application (middle) Tier

The application tier is isolated in its own subnet. There are multiple virtual machines set up for fault tolerance and easy patch management. These VMs can be backed by shared storage, which is offered by Azure NetApp Files (ANF) and/or Premium SSDs. This configuration allows for easier deployment of patches without downtime. The machines in the application tier should be fronted by a public load balancer so that requests to the EBS application tier are processed even if one machine in the tier is offline due to a fault.

Load balancer

An Azure load balancer allows you to distribute traffic across multiple instances of your workload to ensure high availability. In this case, a public load balancer is set up, because users are allowed to access the EBS application over the web. The load balancer distributes the load to both machines in the middle tier. For added security, allow traffic only from users accessing the system from your corporate network using a site-to-site VPN or ExpressRoute and network security groups.

There are a significant number of configurations and HA options for an Azure load balancer that can support various application configurations and requirements. If a load balancer doesn't meet the needs of the application, an application gateway or an Azure Route Server.

Database tier

This tier hosts the Oracle database and is separated into its own subnet. It is recommended to add network security groups that only permit traffic from the application tier to the database tier on the Oracle-specific database port 1521.

Microsoft and Oracle recommend a high availability setup. High availability in Azure can be achieved by setting up two Oracle databases in two availability zones with Oracle Data Guard. Clearly understand the difference between how we architect for a cloud environment and the choices made for an on-premises datacenter solution. Where RAC may justify in an on-premises data center, it tends to be much less valuable in a 3rd party cloud. Even if it was useful, Oracle will not support RAC in any public cloud. On top of this, the Azure cloud High Availability (HA) architecture solutions are often in contradiction with what RAC offers, creating a nonsensical solution.

1. RAC is often a marketing opportunity for Oracle. RAC must be acknowledged as an instance resiliency and scalability product, often not meeting many basic HA requirements. It is A solution, not THE solution and rarely do find workloads that require it for scaling, as well as benefit on savings on resources and price for customers once cloud architectural differences are realized.

2. Oracle only supports RAC in Oracle Cloud or on-premises and will refuse support in any third-party cloud environment, including Bare Metal.
3. Choose Oracle Data Guard for DR and HA, as it is very complementary to Azure HA design, just as Always-on AG is for SQL Server. We deploy the DG Broker, the observer and configure Fast-Start Failover to automate any failovers and manual switchovers and the DBMS_Rolling package will allow for online patching and upgrading.

Performing a Sizing Assessment with the AWR

Disclaimer: *Each version and database type of the Automatic Workload Repository (AWR) report can display data differently. The fields are the same, but the data may be in a different order, have a different header, etc. This document is to offer guidance in filling it out. If unsure, escalate for assistance, as an incorrect number could impact sizing estimates if not performed correctly.*

Assumptions

- AWR Report with 1-day or longer workload report
  - Ideally the report should cover peak load times
- The AWR Analysis sizing template
- Basic understanding of AWR data and Excel
- The Oracle database is either a single Oracle instance or RAC
- The Oracle database isn't on an engineered system such as Exadata

Links to Worksheet

The worksheet template is in the following GitHub repository.:

Oracle AWR to Azure IaaS Worksheet: OracleOnAzure/Oracle_AWR_Estimates.xltx at master · Dbakevlar/OracleOnAzure · GitHub

Detailed Instructions: OracleOnAzure/AWR Sizing Instructions.docx at master · Dbakevlar/OracleOnAzure · GitHub

Updates to the worksheet are made regularly to fulfill requests by Oracle customers in Azure and some changes may be present vs. the examples shown in this documentation included in this paper.

Process

Although the AWR report can provide essential data about workload, database usage and optimization for a cloud project, specific calculations can offer us invaluable data on what is required for an Azure IaaS VM to run the Oracle database in the cloud. The following will explain step by step what values to gather from the report and where to place them in the spreadsheet.

The Spreadsheet is broken down into two worksheets, the AWR and the Calculations worksheet. There are multiple lines to take RAC and multiple instances into consideration.

The AWR Worksheet

The first three columns:

DB Name: the unique name given to the database.

Database		
Id	Name	RAC
1633071752	GBACSPRP	YES

Instance Name: Is the same as individual database node names in RAC or often the same as the DB Name for non-RAC databases.

Host Name: The name of the host. For RAC, each node will have a unique name.

Instance	Host
gbacsprs1	gbslo1exdb01.dunnhumby.c

Elapsed Time and DB Time: These two sections are commonly next to each other throughout the report.

Report Total (minutes)	
DB time	Elapsed time
14,628.65	10,040.28

DB CPUs: This can be a confusing metric, as CPU data is in numerous fields, but the value we're searching for is referred to as "DB CPU(s)". Enter it for each instance involved in the estimate.

Foreground Wait Classes

I#	User I/O(s)	Sys I/O(s)	Other(s)	Applic (s)	Commit (s)	Network (s)	Concurcy (s)	Config (s)	Cluster (s)	Admin (s)	Scheduler (s)	DB CPU (s)	DB Time
1	153,108.51	1,576.73	37,136.16	123,943.11	24,202.46	19,002.52	115,801.06	41,247.35	110,284.78	41.40	1,401.77	1,463,292.19	2,048,549.15
2	230,252.91	630.03	33,792.35	101,341.43	76,691.29	13,365.43	52,234.05	66,253.74	80,812.56	11.58	3,097.45	1,476,583.93	2,092,224.28
Sum	383,361.42	2,206.76	70,928.51	225,284.54	100,893.75	32,367.95	168,035.11	107,501.09	191,097.34	52.98	4,499.22	2,939,876.11	4,140,773.43
Avg	191,680.71	1,103.38	35,464.25	112,642.27	50,446.87	16,183.97	84,017.56	53,750.55	95,548.67	26.49	2,249.61	1,469,938.06	2,070,386.71
Std	54,549.33	669.42	2,364.43	15,981.81	37,115.20	3,986.02	44,948.67	17,682.19	20,840.01	21.09	1,199.02	9,398.68	30,882.98

CPUs/Cores: Hyper-threading makes it important to have both these numbers. We commonly calculate off the Cores value and ensure that you update the CPU calculation for it in the spreadsheet if you do note that there is hyperthreading involved. For the example below, a 3-node RAC has 320 hyperthreaded CPUs, with 160 CPU cores total for each.

Always check to verify the VM SKU chosen is using a hyperthreaded vCPU to ensure the calculation for core licensing the customer will bring over to Azure is correctly calculated. If hyperthreading isn't on or has been turned off by Microsoft Support, then licensing from on-premises to Azure is a simple 1:1 cost per Document 2688277.1 (oracle.com):

DETAILS

For the purposes of this document, Non-Oracle Public Cloud Environments are defined as:

- (a) Non-Oracle Public Clouds. Examples: Google Cloud Platform, Amazon AWS, Microsoft Azure, IBM Cloud, Alibaba Cloud, etc.
  or
- (b) Environments that are in any way considered an extension of Non-Oracle Public Clouds including but not limited to running Non-Oracle cloud management software, cloud billing, cloud support, cloud automation, cloud images, or cloud monitoring. Examples: Google Bare Metal Solution, Amazon AWS Outpost, Microsoft Azure Stack, IBM Bluemix Local, Alibaba Hybrid Cloud, etc.

Support Policy for Non-Oracle Public Cloud Environments

Oracle has not certified any of its products on Non-Oracle Public Cloud Environments. Oracle Support will assist customers running Oracle products on Non-Oracle Public Cloud Environments in the following manner: Oracle will only provide support for issues that either are known to occur on an Oracle Certified Platform outside of a non-Oracle Cloud Environment (Oracle Certification Home), or can be demonstrated not to be as a result of running on a Non-Oracle Public Cloud Environment.

If a problem is a known Oracle issue, Oracle support will recommend the appropriate solution on an Oracle Certified Platform outside of a non-Oracle Cloud Environment. If that solution does not work in the Non-Oracle Public Cloud Environment, the customer will be referred to the Non-Oracle Public Cloud vendor for support. When the customer can demonstrate that the Oracle solution does not work when running on an Oracle Certified Platform outside of a non-Oracle Cloud Environment, Oracle will resume support, including logging a bug with Oracle Development for investigation if required.

If the problem is determined not to be a known Oracle issue, we will refer the customer to the Non-Oracle Public Cloud vendor for support. When the customer can demonstrate that the issue occurs when running on an Oracle Certified Platform outside of a non-Oracle Cloud Environment, Oracle will resume support, including logging a bug with Oracle Development for investigation if required.

Support Policy for Oracle Real Application Clusters (RAC)

Oracle does not support Oracle RAC or Oracle RAC One Node running on Non-Oracle Public Cloud Environments.

I#	Num CPUs	CPU Cores
1	320	160
2	320	160
3	320	160

Memory (GB): Memory is captured in the same line as CPU information, but it is calculated differently than we need in our spreadsheet. Remember to convert from MB to GB as part of the steps when you enter the info.

Memory (M)
6,191,158.41

/ 1024= Correct Value for Spreadsheet

%Busy CPU: This value is clearly stated in the report and is used to identify CPU saturation. A CPU is either on or off, but to know if enough CPU is available is part of our estimates. This is another value that can be confusing to gather. Go to the OS Statistics and for each instance CPU totals, look for %Busy.

% Busy
25.84

SGA(MB): This can be under different tables, depending on the version. It can be a good idea to do a search for "SGA". SGA Target demonstrates the beginning and end values for an adjusting value. If you use this section, take the highest of the two values, (peak). If no value is shown for an ending value, it means no adjustment was made from the beginning value.

Sga Target	
Begin	End
32,768	

PGA(MB): Is the Process Global Area and this is a specialized area of memory allocated for sorting, hashing and other important processing. Heavier sorting is performed in Oracle due to lacking clustered indexes in the Oracle design. The memory allocated may not meet the needs of the database, which is a resiliency vs. sizing issue. Like SGA, the PGA Target will display a beginning and ending value for some AWR Reports. Take the larger of the two values displayed.

PGA Target	
Begin	End
32,768	

Read Throughput (MB/s) and Write Throughput (MB/s): This is a value that can be displayed in multiple ways and sections in the AWR report depending on the version and type of Oracle product. Search the report, (find on page if in a browser) for "IO Statistics". For the example below, a RAC database with 3 nodes displays the Read throughput and write throughput for each instance:

IOStat by Function

		Requests				MB					
I#	Function Name	Total	IOs/s	Reads/s	Writes/s	Total	IO MB/s	Read MB/s	Write MB/s	Waits: Count	Avg Time
*	Total	1,581,621,029	29,293.53	19,839.48	9,454.06	452,920,597	8,388.64	6,568.84	1,819.80	519M	507.38us
	RMAN	117,080,157	2,168.47	1,895.60	272.86	126,061,919	2,334.82	1,790.06	544.76	294K	.99ms
	Smart Scan	120,087,660	2,224.17	2,224.17	0.00	100,626,420	1,863.72	1,863.72	0.00	0	
	Others	142,617,748	2,641.45	2,347.72	293.73	70,587,460	1,307.37	1,035.42	271.95	81M	772.34us
	Streams AQ	65,466,095	1,212.51	1,212.51	0.00	68,204,442	1,263.23	1,263.23	0.00	876K	6.88ms
	LGWR	79,729,417	1,476.69	0.13	1,476.55	27,737,403	513.73	0.00	513.73	12.4K	818.45us
	Direct Reads	33,626,408	622.80	553.19	69.61	17,107,441	316.85	282.22	34.63	0	
	Buffer Cache Reads	624,211,846	11,561.16	11,561.16	0.00	16,889,994	312.82	312.82	0.00	436.7M	445.09us
	Direct Writes	56,913,309	1,054.10	44.95	1,009.15	14,721,412	272.66	21.36	251.29	8979	640.09us
	DBWR	341,886,723	6,332.16	0.01	6,332.14	10,984,057	203.44	0.00	203.44	645	573.64us
	Data Pump	1,666	0.03	0.03	0.00	49	0.00	0.00	0.00	934	676.66us
1	Total	726,133,793	13,448.87	9,231.35	4,217.53	248,993,562	4,611.66	3,738.47	873.19	204.9M	522.33us
	RMAN	70,066,122	1,297.71	1,144.31	153.40	74,193,226	1,374.15	1,067.96	306.19	177.3K	.99ms
	Smart Scan	68,800,451	1,274.27	1,274.27	0.00	60,816,059	1,126.39	1,126.39	0.00	0	
	Streams AQ	39,765,909	736.51	736.51	0.00	41,358,808	766.02	766.02	0.00	523.2K	6.07ms
	Others	62,115,807	1,150.46	1,027.04	123.42	32,428,062	600.61	486.51	114.10	33.8M	735.56us
	LGWR	32,315,706	598.53	0.07	598.45	12,034,230	222.89	0.00	222.89	6855	829.32us
	Direct Reads	17,122,289	317.13	284.93	32.20	8,617,683	159.61	142.46	17.15	0	
	Buffer Cache Reads	257,165,089	4,763.01	4,763.01	0.00	7,998,854	148.15	148.15	0.00	170.4M	462.48us
	Direct Writes	22,510,719	416.93	1.17	415.75	6,333,887	117.31	0.99	116.32	180	2.00ms
	DBWR	156,270,035	2,894.31	0.00	2,894.31	5,212,704	96.55	0.00	96.55	258	810.08us
	Data Pump	1,666	0.03	0.03	0.00	49	0.00	0.00	0.00	934	676.66us
2	Total	855,487,236	15,844.66	10,608.13	5,236.53	203,927,035	3,776.98	2,830.36	946.61	314.1M	497.63us
	RMAN	47,014,035	870.76	751.29	119.46	51,868,693	960.67	722.10	238.57	116.7K	.98ms
	Smart Scan	51,287,209	949.90	949.90	0.00	39,810,361	737.34	737.34	0.00	0	
	Others	80,501,941	1,490.99	1,320.68	170.31	38,159,398	706.76	548.91	157.85	47.2M	798.65us
	Streams AQ	25,700,186	476.00	476.00	0.00	26,845,634	497.21	497.21	0.00	352.8K	8.08ms
	LGWR	47,413,711	878.16	0.06	878.10	15,703,173	290.84	0.00	290.84	5522	804.96us
	Buffer Cache Reads	367,046,757	6,798.15	6,798.15	0.00	8,891,140	164.67	164.67	0.00	266.4M	433.97us
	Direct Reads	16,504,119	305.68	268.26	37.41	8,489,758	157.24	139.76	17.48	0	
	Direct Writes	34,402,590	637.18	43.78	593.40	8,387,525	155.35	20.37	134.98	8799	612.34us
	DBWR	185,616,688	3,437.85	0.01	3,437.84	5,771,353	106.89	0.00	106.89	387	416.02us

Read IOPs/Write IOPs: Like throughput, this section can be displayed in different parts of the AWR report, but often is in the Load Profile towards the top of the report or in the IO Statistics in the mid-section of others.

System Statistics - Per Second

I#	Logical Reads/s	Physical Reads/s	Physical Writes/s	Redo Size (k)/s	Block Changes/s	User Calls/s	Execs/s	Parses/s	Logons/s	Txns/s
1	1,659,628.69	95,997.50	14,981.18	110,307.98	129,308.78	2,156.57	1,713.20	638.67	11.91	85.91
2	1,677,125.77	78,415.62	17,547.20	144,005.17	123,147.77	1,635.36	1,660.35	1,079.14	13.17	80.84
Sum	3,336,754.45	174,413.12	32,528.39	254,313.14	252,456.56	3,791.93	3,373.55	1,717.81	25.09	166.75
Avg	1,668,377.23	87,206.56	16,264.19	127,156.57	126,228.28	1,895.96	1,686.77	858.91	12.54	83.38
Std	12,372.31	12,432.27	1,814.45	23,827.51	4,356.49	368.55	37.37	311.46	0.89	3.58

Calculating Factors for Worksheets

Once you've filled in this information, note that there is a gray box below the area to enter in all sections, for instance:

Peak CPU factor	3.00
Est'd RAM factor	1.50
vCPU HT factor	2.00
Busy CPU waits factor	2.00
IO metrics (IOPS & MB/s) fudge factor	1.50

These values are here to help calculate the type of workload that you are bringing over. For Exadata, an IO metric fudge factor would be high, (in the example, 6 times what is being experienced in the workload) to take increased IO into consideration from loss in offloading and other engineered features.

Decide what you want for each of the following and make changes based on the following:

Peak CPU Factor: 2.00 is standard, 4.00 is for a workload that might have a huge variance expectation once it goes to the cloud.

Est'd RAM Factor: Same for CPU, but for RAM estimate. Normal is 2.00, 4.00 would be normal for an Exadata where the SGA is commonly shrunk to promote offloading.

vCPU HT Factor: Commonly 2.00 and this should be the default going to IaaS Azure VMs

Busy CPU waits factor: 2.00 is the default

IO metrics (IOPS & MB/s) fudge factor: 2.00 is for transactional system, 4.00 is for DSS/OLAP, 6.00 is for Exadata.

Calculations Spreadsheet

*Don't fill in any area OUTSIDE of the fields instructed, which have headers **filled with blue**. Columns are dependent on what is filled in on the AWR page to match what is in the appropriate fields on the Calculations page.*

Enter the DB Name and Instance Name, duplicating the DB name, if necessary, that corresponds to the instance name. Do not leave the first column blank if you fill in the second.

DB Name	Instance Name

Although the column looks like it extends for two, place the hostname for the servers for every instance in the first column of the next section.

Host	Name
Total	

Enter the DB name, (database name, NOT instance name) in the third section, first column. If working in a RAC environment, the RAC database will be listed by the DB name one time, not for each node in this section. The Excel spreadsheet will calculate and total the resources required for a **single instance,** as this is our primary goal to achieve a fully supported environment on Azure by Oracle.

DB Name
Total

As you're entering the values into this second worksheet, calculations will appear. Once complete, you should have values for each database to size the workload into Azure. These values will then give you the information you need to choose one or more IaaS Azure VMs to size out a solution for the Oracle customer.

Example of Calculations for RAC to Single Instance

The following is an example of the output from a customer engagement. This involves two databases, both 2-node RAC environments. Notice that the DB Name column is listed twice for both, then the instance name is unique. No other information was filled in, as the value in the previous worksheet automatically populates and calculates what is needed.

Calculated detail by database instance							
DB Name	Instance Name	%DB Time of Elapsed Time	%DB CPU of server capacity	Total ORA (GB)	Total IOPS	Total Throughput (MB/s)	Est'd Azure vCPUs
ARPH2PPD	ARPH2PPD1	3059.755%	48.951%	310	1,296.50	125.69	61.20
ARPH2PPD	ARPH2PPD2	2915.155%	45.965%	310	1,433.08	127.69	58.30
ARPH2PRD	ARPH2PRD1	669.554%	8.185%	310	2,538.29	190.88	13.39
ARPH2PRD	ARPH2PRD2	679.164%	8.366%	310	3,021.84	276.57	13.58
		7323.628%	111.469%	1,240	8,289.71	720.83	146.47

In the second section, only the host's name was populated to the first column for each of the nodes for the RAC instances. As there are two nodes each for the two databases, four entries are added, and the values populate from the first worksheet.

Aggregated calculations by host		
Host	Name	%DB Ti
dbsls504		
dbsls505		
dbslp2030		
dbslp2031		

In the last section, I only listed the two global database names. The data for each of the nodes for each of the databases is calculated and total resources are displayed for the environment to be moved to Azure IaaS VMs. With the factoring numbers taken into consideration, we have average workloads from the AWR and then peak workloads which are calculated from the workloads and the factoring numbers.

Aggregated calculations by database									
DB Name	%DB Time of Elapsed Time	Total vRAM (GiB) consumed only by Oracle	Est'd Azure vRAM for server	Total IOPS	Total Throughput (MB/s)	Est'd Azure IOPS for peak load	Est'd Azure Throughput (MB/s) for peak load	Est'd Azure vCPUs for avg load	Est'd Azure vCPUs for peak load
ARPH2PPD	2987.455%	620	930	2,729.58	253.38	4,094.37	380.07	32	84
ARPH2PRD	674.359%	620	930	5,560.13	467.45	8,340.20	701.18	24	64
Total	3661.81%	1,240	1,860	8,289.71	720.83	12,434.57	1,081.25	56	148

For example:

ARPH2PPD will require:

- 32 vCPU for an average load and 84 vCPU for a max workload.
- A server with 930G of memory and 620G allocated to the database.

- Disk IOPS 4094 and 380MB/s throughput

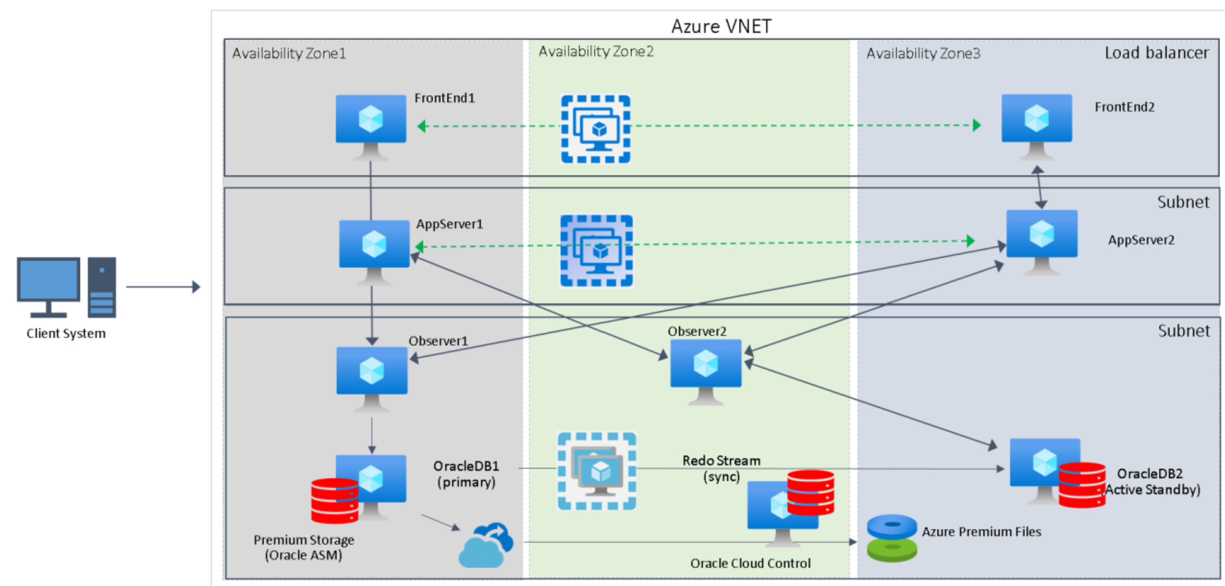
Calculations can be seen for the second database to be migrated to a single instance, ARPH2PRD.

There is a total that is displayed at the bottom, but this is only available if you need to know how many resources will be required for the project. The value we have here is what we require to size out the Azure VM.

Choosing the Correct VMs and Storage

High Level Oracle on Azure for IaaS

Oracle High Availability in the Azure cloud marries Azure High Availability with Oracle Data Guard to create solutions that use many of Oracle's Maximum Availability Architecture advanced concepts. Due to the differences between on-premises architecture and the public cloud, there are significant differences. Where Data Guard is more focused on Disaster Recovery in an on-premises solution, in Azure, it's front and center for High Availability, leveraging Fast-Start Failover, the DG Broker, Observer, etc.



Decisions around Cross-region deployments, Availability Zones, or Availability Sets, along with number of Data Guard standbys in a specific customer environment is based on Service Level, Recovery Point Objective, (RPO) and RTO, (Recovery Time Objective). This information will also provide the information required for backup and recovery strategies and storage requirements, (storage often has features that provide value in these focus areas.)

The above, classic highly available and 99.996% uptime for Oracle is a recommendation to begin with and from here, the architecture can simplify or evolve.

Azure recommendations for Oracle Virtual Machines

Below are some typical Oracle VM configuration checklist items

Type	Source	Azure Recommendation
Storage	https://docs.microsoft.com/en-us/azure/virtualmachines/windows/premium-storage#scalabilityand-performance-targets	Use Premium storage for data files with size P40 or P50 unless workload MBPs, (throughput) requires more than the attached storage can supply, (see storage matrix)
	https://docs.microsoft.com/en-us/azure/virtualmachines/workloads/oracle/oracle-design#diskcache-settings	Separate redo logs from datafiles TS on separate data disk whenever possible.
	https://docs.microsoft.com/en-us/azure/virtualmachines/linux/optimization#io-schedulingalgorithm-for-premium-storage	Set NOOP or Deadline algorithm for I/O scheduling
	https://docs.microsoft.com/en-us/azure/virtualmachines/windows/premium-storage#premiumstorage-for-linux-vms	Disable "barriers" for disks with cache readonly or none, (see caching options by purpose below in this table)
	https://docs.microsoft.com/en-us/azure/virtualmachines/windows/premium-storageperformance#disk-striping	Use Stripe size 64KB

Type	Source	Azure Recommendation

	https://docs.microsoft.com/en-us/azure/virtualmachines/workloads/oracle/oracle-design	For OS disks, use default Read/Write host level caching and use premium SSD, (P10 or P15 recommended for an Oracle VM)
Temp/Swapfile	Ephemeral OS disks - Azure Virtual Machines | Microsoft Docs	Ensure that swapfile for Linux or Windows is located on attached, ephemeral storage on VM whenever possible. Monitor the choice in VM that it includes temp storage.
	https://docs.microsoft.com/en-us/azure/virtualmachines/workloads/oracle/oracle-design	For DATAFILES, use Read-Only host level caching for Premium SSD. P40 or P50 is the preferred premium SSD of choice that is Read-Only host level caching capable. For the P50, don't allocate the last 1G to stay under the max size of 4095G for host level caching.
Redo logs	https://docs.microsoft.com/en-us/azure/virtualmachines/workloads/oracle/oracledesign#configuration-options	Separate Redo logs from other data files
	Ultra disks for VMs - Azure managed disks - Azure Virtual Machines | Microsoft Docs	Use Ultra Disk, which is price effective for high IO demands and redo latency. Scaling feature as redo demands increase.
	https://docs.microsoft.com/en-us/azure/virtualmachines/linux/how-to-enable-write-accelerator	Enable Write Accelerator for Redo logs disks
	https://docs.microsoft.com/en-us/azure/virtualmachines/linux/how-to-enable-write-accelerator	Set Cache policy: None + Write Accelerator for Redo logs disks
	https://docs.microsoft.com/en-us/azure/virtualmachines/linux/how-to-enable-write-accelerator	Use I/O sizes (<=32 KiB) (Redo block size < 32)

Network	https://docs.microsoft.com/en-us/azure/virtualnetwork/create-vm-accelerated-networking-cli	Use Accelerated Networking, dependent upon VM SKU choice for availability.
Oracle DB		if filesystem is ext4 use DB param:
		filesystemio_options=ASYNCH
		If using ASM, partitions used for ASM disks should be created with a 1MB (2048 sectors) offset
		If using ASM, set diskgroup au_size >= 4M for large databases
	https://docs.oracle.com/database/121/UNXAR/appi_vlm.htm#UNXAR391	Use ASMM, Enable HugePages on Linux and disable Transparent HugePages: https://oraclebase.com/articles/linux/c onfiguring-huge-pagesfor-oracle-on-linux-64 + HugePages on Oracle Linux 64-bit (Doc ID 361468.1)

VCPU is the least of your worries

Unlike on-premises, high vCPU count and memory is available in the Azure Cloud. Rarely do Oracle workloads run into a challenge on achieving the vCPUs required after a sizing assessment. As Oracle charges licensing by core, this is another reason to perform a right-sizing assessment before recommending a VM SKU for the Oracle database workload.

High Memory Shouldn't be the Default for Oracle

Azure VM SKUs have four main categories for relational workloads:

- General Purpose
- Compute Optimized
- **Memory Optimized**
- Storage Optimized

Of these, only **Memory Optimized** are primarily used for Oracle workloads. The Memory Optimized VM SKU category includes SKUs from the D, E and M series virtual machines. For Oracle, as discussed earlier in the recommended checklist, there are two areas that we drill down on for preferred SKU series:

- E-series, Eds v4 or v5 is highly recommended
    - Allows for Premium SSD for OS Disk
    - Has ephemeral storage to be used for swap
    - V4 is readily available in all regions
    - V5 has great performance enhancements, but limited availability
    - High IO limits
- M-series
    - Great for high memory requirements
    - Enhanced vCPU performance
    - Lower IO limits, host level caching is low for attached storage vs. E-series
    - Has ephemeral storage to be used for swap
    - Offers accelerated networking options

The Oracle database performances are strictly influenced by the following parameters:

- Disk throughput, (MBPs)
- Read/write IOPS
- Network latency
- CPU, RAM

High IO Storage Matrix

Throughput, (MBPs) is the most important aspect of sizing an Oracle database when migrating to the Azure cloud. Standard SSD won't support these high IO workloads and knowing when to evolve to the next level of storage is important. The matrix below covers the most common challenges that customers will want answers to when deciding on what storage, including options with third-party storage that few are aware are available to run Oracle on Azure. Storage certification/support isn't required with Oracle, (Oracle with SAP does have requirements when running on Azure, which means currently, only Premium SSD and ANF is supported).

Workload	Premium SSD	Ultra Disk	Azure NetApp(ANF)	Silk	Flashgrid
IO up to 750MBPs	X	* Rarely used, not cost effective	X	X	X
IO up to 2000MBPs		X	X	X	X
IO up to 5000MBPs			X	X	X
IO 10K MBPs +				X	*with RAC
Exadata Offloading with HCC				X Silk Compression	* With RAC can compensate
Non-Platform specific	X	X	X	X	Oracle Only
RAC Capable			* On RAC Bare Metal	X	X
Check Region Availability		X	X	X	X
Non-AZ Deployment			X	X	
Limited Use Cases		* Mostly used with Redo Logs			
Thin Snapshots	X	Private Preview	X	X	
Thin Clones				X	
Scaling Storage Options	X	X	X		X
Enterprise Cost	$	$$$	$$$$	$$$	$$$

Storage considerations

When you create a new managed disk from the portal, you can choose the Account type for the type of disk you want to use. Keep in mind that not all available disks are shown in the drop-down menu. The lowest performing storage to be used for Oracle workloads recommended is Premium SSD, followed by Premium SSD with Ultra disk support for redo logs, then scaling to Azure NetApp Files and to third-party solutions. The matrix offers a high-level look at feature comparisons so that the Oracle workload will help decide what is the best fit. After you choose a particular VM size, the menu shows only the available premium storage SKUs that are based on that VM size.

- After you configure your storage on a VM, you might want to load test the disks before creating a database. Knowing the I/O rate in terms of both latency and throughput can help you determine if the VMs support the expected throughput with latency targets.

- There are several tools for application load testing, such as SLOB, (Silly Little Oracle Benchmark, Oracle Orion, Swingbench, and FIO. Due to lack of community support, the open-source product of HammerDB is less recommended for IO testing and TPM, (Transactions per minute) aren't available for platform comparisons like they are for SQL Server or MySQL.

- Run the load test again after you've deployed an Oracle database. Start your regular and peak workloads, and the results show you the baseline of your environment.

- Focus should always be given on MBPs, (throughput) followed by IOPs vs. the storage size. For example, if the required MBPs are 750, but you only need 200 GB, you might still get the P40 class premium disk even though it comes with 1000 GB of storage.

This way you can meet the MBPs requirement, (if host level read-only caching it turned on).

- The MBPs/IOPS rate can be obtained from the sizing assessment done from the AWR report. It's determined by the redo log, physical reads, and writes rate gathered from the AWR, but aggregations and averages are taken into consideration for larger window workloads and assessed in the end values to give a more realistic view of peak, IO workloads.

High IO Storage Solutions

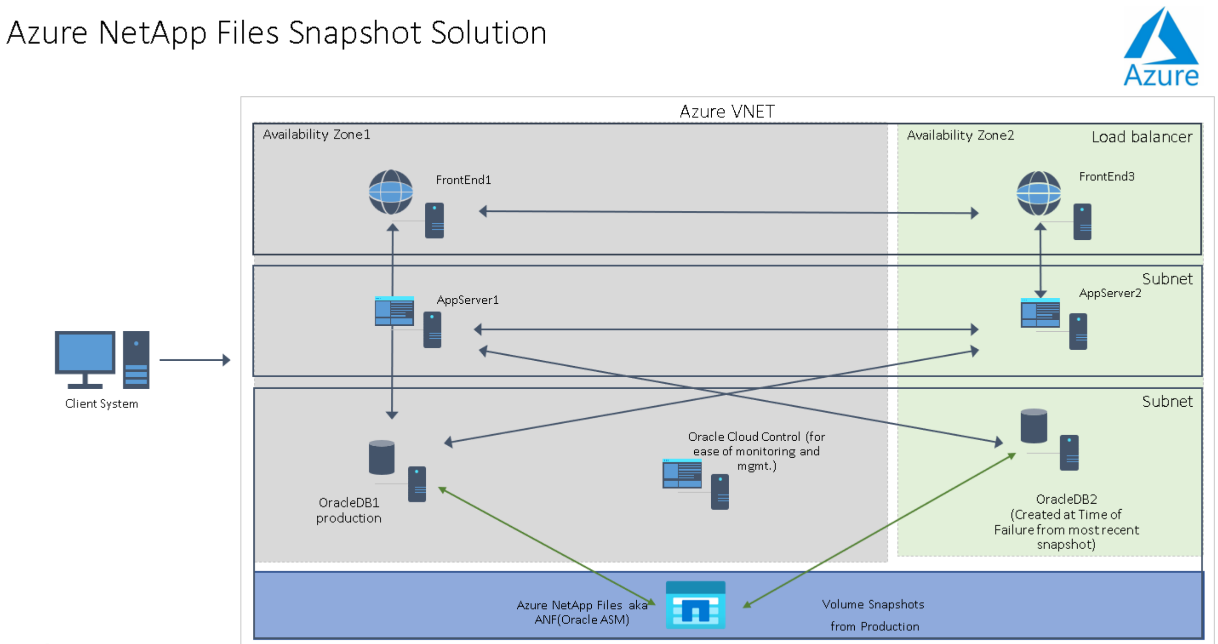
Although there are numerous recommendations for running Oracle on Azure long-term, having the throughput these high IO workloads require is one of the most important keys to success. Along with high IO native solutions, like Azure NetApp Files, (ANF) there's also third-party solutions to consider- even Oracle Exadata workloads can successfully run on Azure when architected to gain the best from Azure cloud.

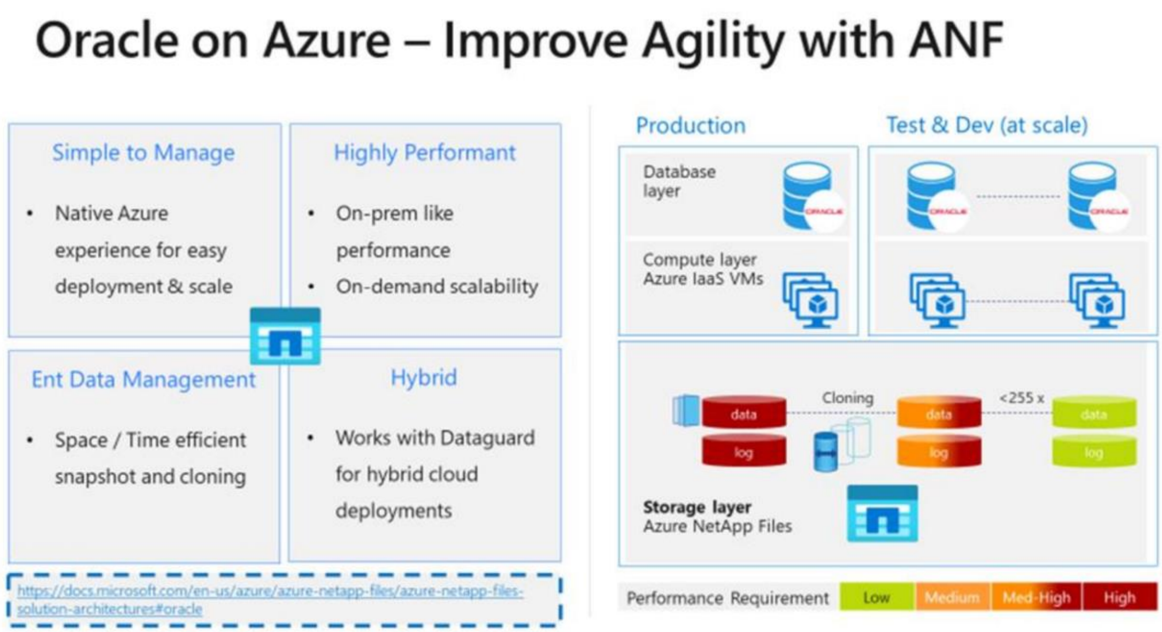
As Oracle has stopped requiring certification to provide support at the storage layer, (except for unique software deployments like SAP with Oracle) these solutions are no different than any other storage solution when running Oracle on Azure.

Azure NetApp Files

Unlike the name might suggest, Azure NetApp Files, (ANF) is an Azure first, aka native solution inside the Azure cloud. Designed in partnership with NetApp, ANF is an incredibly flexible and scalable storage option for high IO workloads that removes much of the manual configuration for IaaS deployments with a simple service option for deployments at the speed of the cloud.

Unlike an attached storage solution, such as premium SSD or Ultra Disk, ANF is a network attached solution, available in a capacity pool that can be used by multiple databases and VMs across numerous Availability Zones.



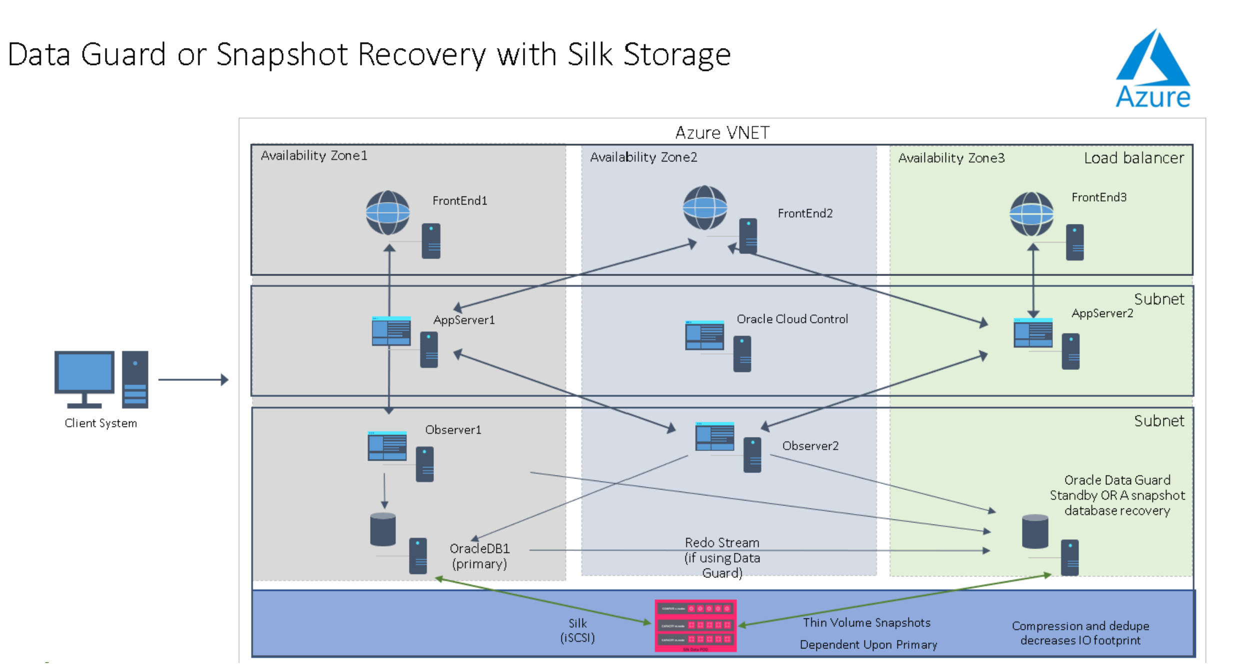


ANF comes in standard, premium and ultra for their capacity pools and can be scaled up and down as the customer needs which is a feature that many customers really appreciate as workload demands change in their Oracle database.

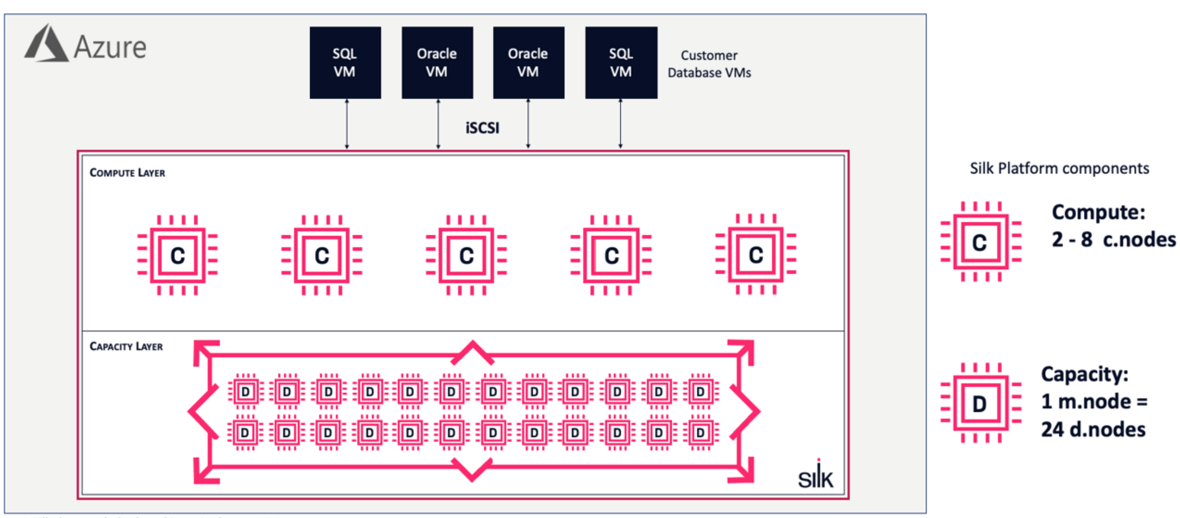
Silk

Although Silk is a third-party solution, it's all Azure under the covers for hardware. A Silk data pod mirrors the architecture of traditional on-premises SAN but using D-series for controllers and L-series

Azure Virtual Machines as the disk arrays. These are configured as part of a Kubernetes data pod that provides high IO performance using the ephemeral disk attached to the VMs. With the addition of compression/dedupe, along with thin volume snapshot backups and thin cloning capabilities, this solution provides exceptional throughput at a great price.



Silk Platform Architecture



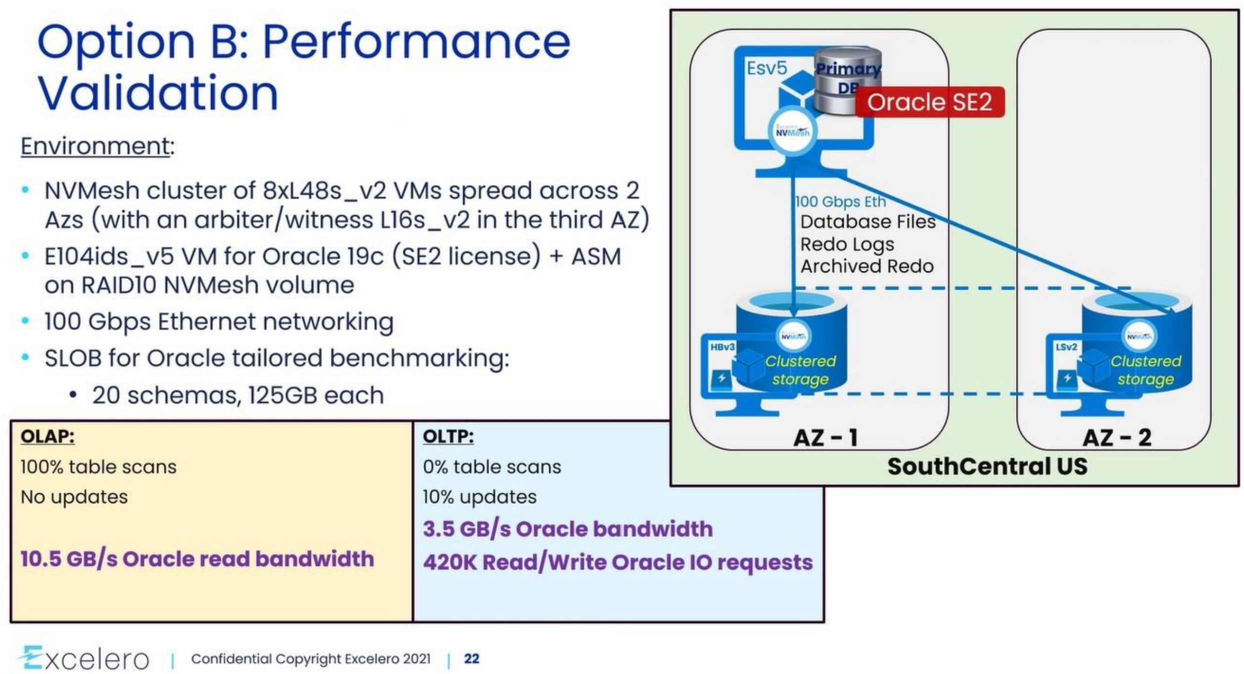
A Silk data pod, deployed in a single zone, RG

The Silk data pod is presented as a storage layer to the database VM and is completely transparent to the database. The Oracle database simply views it as storage to use, just as it would any other disk, but this storage is very fast.

Silk IN THE PUBLIC CLOUD							
c.nodes	2 c.nodes	3 c.nodes	4 c.nodes	5 c.nodes	6 c.nodes	7 c.nodes	8 c.nodes
Usable Capacity*	12TB-0.5PB	12TB-1PB	12TB-1.5PB	12TB-2PB	12TB-2.5PB	12TB-3PB	12TB-3.5PB
IOPS	Up to 220K	Up to 330K	Up to 440K	Up to 550K	Up to 660K	Up to 770K	Up to 880K
Throughput	Up to 3.5GB/s	Up to 5.25GB/s	Up to 7GB/s	Up to 8.75GB/s	Up to 10.5GB/s	Up to 12.25GB/s	Up to 14GB/s
Latency	0.25ms						

Excelero NVMEsh

Excelero, like Silk, uses VMs to create a very fast IO solution from VMs in Azure. The difference is that Excelero uses an Nvmesh protocol and has several VM configurations to meet different requirements for customers. Excelero can currently hit read speeds (20+GBps) that are well beyond other solutions but requires very specific VM SKUs which support InfiniBand/MPI network son the backend – namely the HBv3s at the time of writing – to hit these numbers.

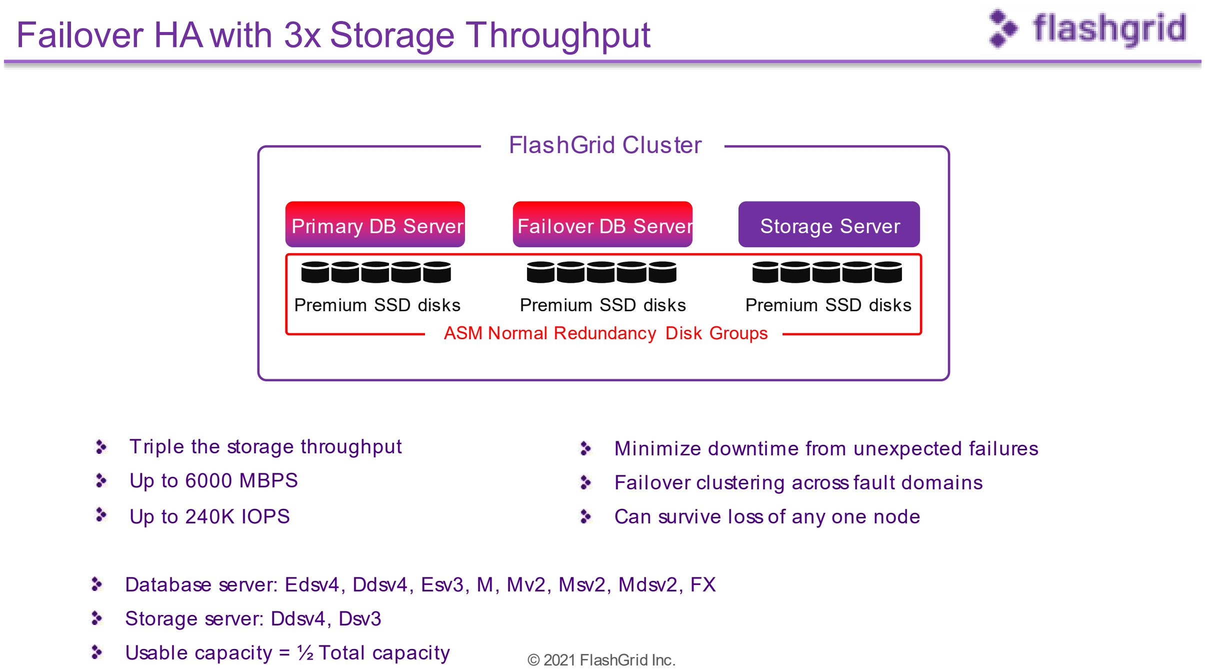


Flashgrid IO

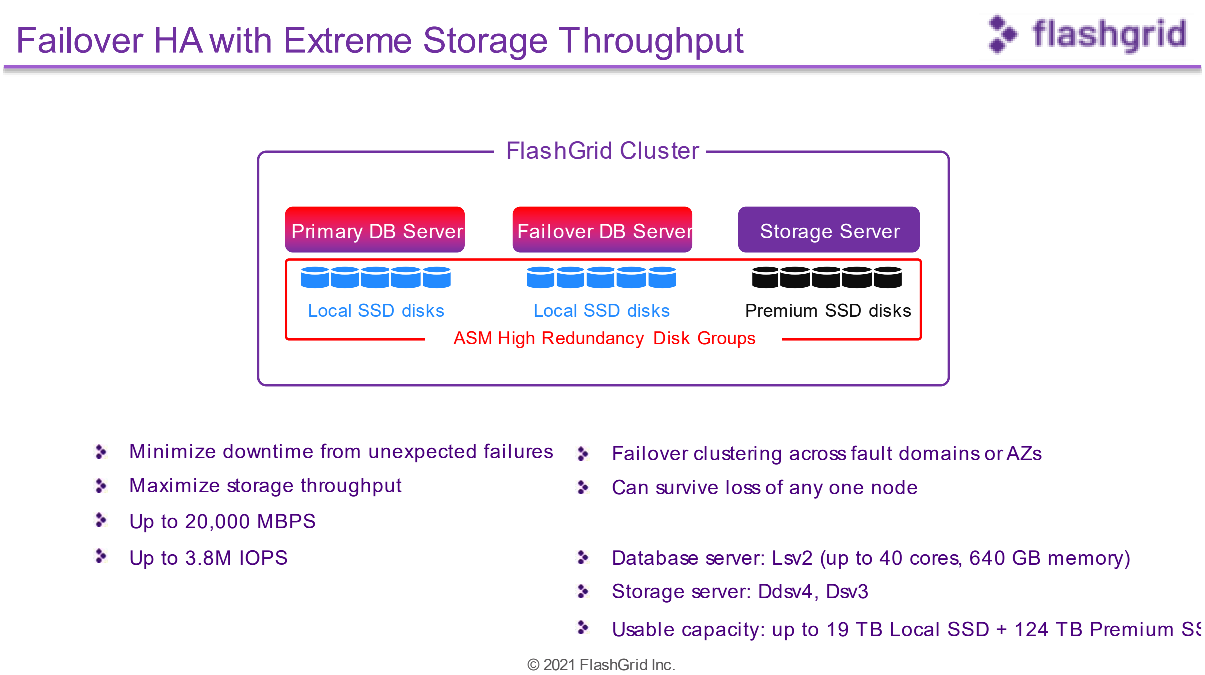
Flashgrid is better known as the company with a RAC solution in Azure, but they also provide a storage solution that is exceptionally fast using collective virtual machines to provide the boost that high IO workloads require.

Like Excelero, Flashgrid has several high IO solutions depending on workload requirements. The main difference for Flashgrid vs. the other solutions above is that Flashgrid is Oracle specific. The other benefit is if the customer is resolute on having RAC in a third-party cloud, Flashgrid has a Real Application Cluster, (RAC) solution and although Oracle won't provide support for it, Flashgrid continues to receive stellar reviews from customers on the support provided by Flashgrid for their clustering solution.

Oracle on Azure using Flashgrid's 3X Storage Solution:



For workloads requiring upwards to 20K MPBs using Flashgrid's Extreme Storage Throughput solution:



When deciding on what solution to use, the storage matrix below can be very beneficial. The correct solution isn't always based on highest throughput or lowest cost, but often on the complete picture of what you're trying to achieve.

Unified identity and access management

With Azure Active Directory, Microsoft extended the features of AD to the cloud enabling single sign-on for enterprise applications and web applications deployed in the cloud. With cross-cloud connectivity, Oracle customers can integrate access management based on Azure Active Directory through a federated identity model. This delivers a unified mechanism for authentication and authorizing users and applications.

The benefits of federated Identity include single sign-on, reduced security risks and increased organizational productivity. Strengthening security posture with Azure

Enterprises can rely on a cloud that is built with customized hardware, has security controls integrated into the hardware and firmware components, plus added protections against threats such as DDoS.

- Take advantage of the multi-layered state-of-art security delivered in Azure data centers globally
- Protect workloads quickly with built-in controls and services in Azure including identity, data, networking, and apps.
- Detect threats early with unique intelligence

Benchmarking

During the initial conversation, the topic of on-premises vs. cloud performance is almost a given. Although there's never a clear apples-to-apples comparison which can be performed between dedicated on-premises hardware and cloud architecture, there are benchmark tools that provide some value in comparing what opportunities lie in cloud migrations for relational workloads.

As often as we collect data around CPU and memory usage, IO is the most valuable indicator on successful migrations to the cloud. Where scaling vCPU and memory is quite easy in cloud environments, storage may not be as simple, especially if Oracle's Automatic Storage Management (ASM) is also part of the solution.

FIO- Flexible IO benchmarking tool was developed by Jens Axboe to enable flexible Linux I/O subsystem and scheduler testing. Having a single testing capability that provided IO performance information, which could be used across all types of applications and simulate workload had great benefit to many administrators, which explains that popularity of the tool, even today.

This tool is contributed to by over 5000 users in the Linux industry and is available to anyone who wants to benchmark various I/O workloads.

- [General documentation on FIO]
- [FIO on Github]
- [FIO Github]
- [FIO Workload Benchmark Examples]

SLOB- i.e., Silly Little Oracle Benchmark, is often the go-to for Oracle specialists to perform Oracle specific benchmarks. It is an open-source tool, maintained by the Oracle community and comes with easy workload generation. If you'd like to know more about SLOB, check out the following links:

- [General Info on SLOB]
- [SLOB GitHub]
- [SLOB Use Cases]

Oracle Swingbench- is an Oracle specific benchmark tool developed by Dominic Giles, who has worked for both Oracle and Google. This tool is very Oracle specific, (as is SLOB) and well-known by Oracle specialists for measuring performance for Oracle workloads.

- [General information on Oracle Swingbench]
- [Swingbench Installation]

Recommended practices with IO Benchmark Tools

1. Expecting the exact same performance in a virtualized environment isn't realistic.

   Identify what response times, network or IO latency is required for the workload as the goal. Be flexible in allocating resources and scaling to meet performance.

2. Use AWR/Statspack reports in conjunction with IO benchmark results

   Most often a performance challenge will have numerous reasons behind the latency. Ensure to identify if Oracle optimizer, maintenance jobs, assumptions on workload are as often the culprit vs. VM and/or storage choices or configurations.

3. Break down performance issues into "consumable" lists.

   Avoid upgrading the database and application tier while migrating to the cloud. Combining multiple projects can cloud performance problems. Always try to separate and tackle each serially than combined.

   If more than one performance challenge, break it down to lists that can be addressed by priority, conquered, and then eliminated.

Migration Recommended Practices

Know Your Database Size

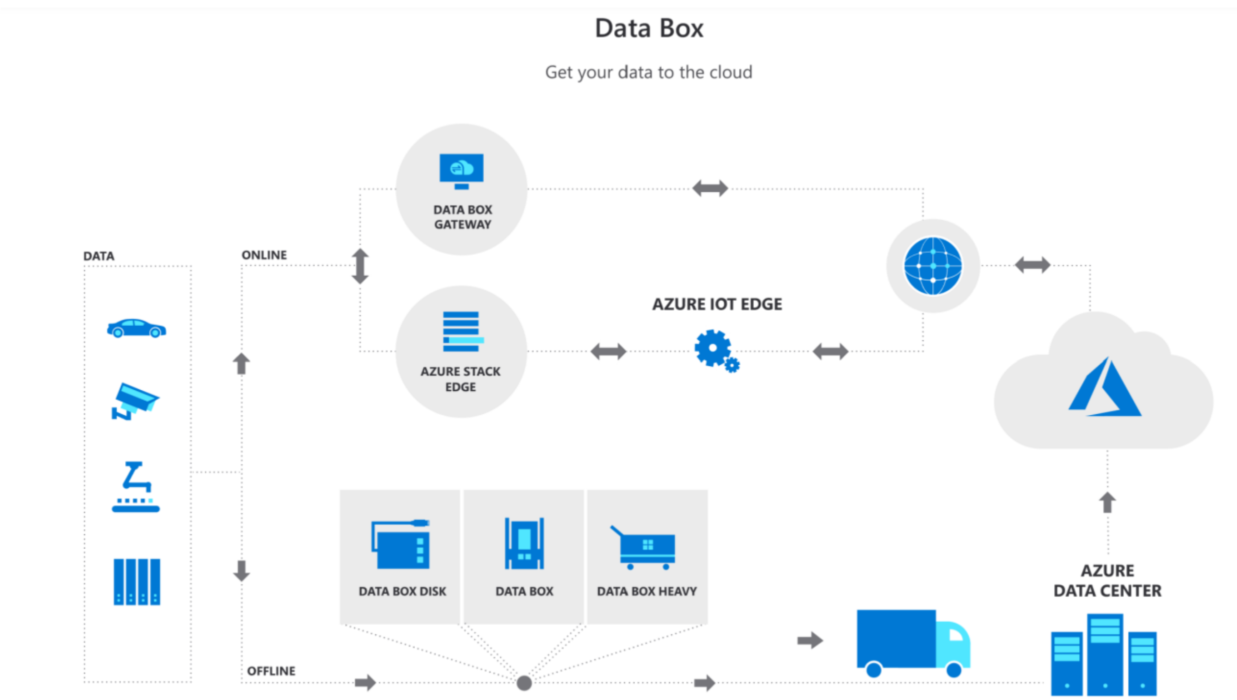
Although you may have heard a significant amount of important information regarding IO, network latency can be an issue in data loading and migrations. The overall size of the Oracle database can be a factor in migration success.

The following script is used to size out the database, identify redo generation, backups, and archive logs, which all dictate the size of the database to be migrated.

Potential Tools for Migrating Oracle to Azure

DataBox- Limited network bandwidth for initial transfers of large Oracle data estates can be a challenge, but with Azure Data Box, customers can use one of three Data Box solutions to provide the right solution to migrate large data workloads to Azure:

- Data Box Disk
- Data Box
- Data Box Heavy



RMAN- Oracle's Recovery Manager is the go-to for Oracle DBAs to backup, recover, and clone databases. This is a comfortable solution for most DBAs, but consideration must be taken that RMAN is a streaming technology that can put heavy IO demands on the network and virtual machines.

Oracle Data Guard With or Without Goldengate – Oracle Data Guard, along with the standard disaster recovery solution for Oracle on Azure, is also a great way to migrate Oracle databases to Azure. With a far sync solution ensuring the changes to the standby running in Azure, a switchover to Azure, making the primary then running in the cloud can be a simple solution to a migration. If a delayed switchover is required, Goldengate can be used in conjunction with a Data Guard environment to simplify the synchronization of the on-premises and cloud environments over time.

Oracle Data Pump – Oracle's import and export tool is a logical backup and recovery tool, but like RMAN, is extremely IO heavy and less performant. All imports are done as inserts and without careful optimization of Data Pump scripting, along with keeping to smaller database workload usage, this tool can deter from meeting migration deadlines.

Hybrid Volume Snapshot Products -

- NetApp CVO, (Cloud Volume Snapshot)
- Commvault Backup and Migration Solutions for Oracle on Azure
- Veeam Backup Solutions in Azure
- Rubrik

Third-Party Synchronization Products –

- Quest Shareplex
- Qlik Replicate
- IBM InfoSphere CDC

Azure Load Balancers – What do load balancers have to do with migration tools? These resources can often help balance out migration workloads and help them migrate more efficiently. Priority can be given to appropriate workloads, letting migrations occur in the background and not overwhelm the resources on virtual environments.

Important Architecture/Processes Related to Migration Success

Azure Express Route- Network bandwidth between the user and the Azure cloud may be essential to solid and consistent performance for users. This will be especially true if the application tier or other aspects of the Oracle environment aren't migrated with the database to the Azure cloud. Although it is always recommended to migrate anything connected to the Oracle ecosystem to Azure with the database, ExpressRoute will offer a more consistent, most stable, lower latency connection to Azure.

Nightly Batch Loads from On-premises – Along with benefiting from Express Route, review any batch loads to the migrating/migrated database for optimization opportunities. Only load what is necessary and recognize that nightly batch loads will be up against Oracle nightly stats collection, along with other maintenance work that is IO heavy in their consumption.

Reporting- Review reports that may perform "SELECT *" or other coding choices that pull more data than required. Unlike batch loads up to the cloud, egress, (data from the cloud) can cause significant cloud cost increases. Some customers have rewritten reporting to decrease consumption from 7TB to 20G with just a few simple, but valuable optimizations in their reporting queries.

Application Location- Always have the goal of keeping the application, middleware, and database tiers in the same availability zone for primary workloads and identify and address any latency issues between the tiers during the POC phase.

Project for Success

Now that you have the tips and tricks to assessing, sizing, and architecture for Oracle on Azure, it's important to build out a set of steps to identify everything that should encompass your project.

1. Identify the database that will be the first to migrate to Azure.
2. Create both an architecture diagram of the on-premises system and an inventory of the systems involved.
3. Review the following and add to the diagram and the inventory:
    a. Schemas in the database, (Each IT group may not connect what they work with by the database name or even application name, so knowing the schema will also help.)
    b. Make a list of all "modules" that are identified connecting into the database. These are the applications and executables. Again, these could go unnoticed by the teams if not inventoried.
    c. Make a list of maintenance jobs and backups.
    d. Add into the inventory the list of nighttime batch jobs or other data load processes.
    e. Identify what are the Recovery Time Objective, (RTO) and Recovery Point Objective (RPO) required for the database and application. This will often decide the type of storage solution for an Oracle on Azure IaaS environment.
4. Collect baseline information from Oracle using Automatic Workload Repositories for busy times in the database to get a clear picture of execution times for the most common SQL and processes, as well as what an average workload looks like.

Building a Proof of Concept

The Proof of Concept (POC) should be designed to verify that the team can run their environment in the cloud and that the team is able to accomplish these migrations with the resources they have on staff. Technical challenges aren't the most common hurdle, but more often it's often lacking knowledge of cloud services or limited time to accomplish the POC. Setting yourself up for success and getting the most out of a POC involves:

1. Create a list of top-ten items that must be tested and will prove the POC's success.
2. Test a REAL workload. Artificial workloads are mostly a waste of time outside of a simple benchmark test using a recommended tool.
3. Choose a mid-range database for the POC, not the largest database to be migrated which has too many complexities and requires too much time and resources to succeed.
4. Minimal complex and fewer application connections but tests a combination of features that are on a top-ten list created by the team.

Switchover Best Practices

Once the POC is completed, recognizing the importance of a successful switchover to the public cloud must be a priority. First and foremost, understanding how critical the application and database are, along with expected downtime for the switchover to the cloud environment will dictate much of how the migration will be architected. While the application tier can be migrated to an Azure VM easily, in comparison, the database often has too many significant block changes to be performed with any Azure service or migration tool. The recommended tools in the Migration section, everything from having a

recovered database in Azure that is being synchronized by Goldengate until the switchover, to those that will use a Data Guard secondary until the switchover is quite common.
Requirements for switchover are:

1. Downtime requirements- a database with 24X7 uptime will not be able to use an outage window for cloud migration downtimes and require a failover in the way of Oracle Data Guard, data replication from on-premises in the form of a change data capture to the Azure cloud or similar.
2. If nighttime data loads are how data is done daily, directing to both the on-premises and the Azure cloud environment could be an option, keeping them synchronized, but each environment is unique and should be identified for opportunities to use what is currently available to make the most of migration success.
3. Use a change control management tool and consider checking in data changes, not just code changes into the system. The database is often the last to be included in change management.

Create Framework for Success

With the use of a Red Hat, (RHEL) or Oracle Linux image from the Azure marketplace for the release version that meets the customer's needs, the Oracle software installation(s) can be built out to an image to be placed in an Image Gallery to be used over and over, eliminating deployment variations and manual work.

This image can then be used as part of the Azure Resource Manager template and be deployed as part of the application framework, simplifying deployment for multiple copies and updates.

Production Optimizing

Once the switchover is complete, it's easy to begin to obsess over performance. Not just as a DBA or infrastructure specialist, but even users may feel that performance seems different and suddenly everyone is a performance expert! This section is to help distinguish where you should focus, when and how.

Inspecting Oracle on Azure Performance

Every Database Administrator has heard this statement, "Nothing's Changed." When there is a migration to the cloud, it could be the same database recovered or cloned to the Azure cloud, but the performance could change drastically due to several things in the existing database:

1. Oracle optimizer settings.
2. Oracle statistics and management plan settings
3. Parameter settings
4. Oracle bug on the cloned database which didn't exist on the original.

The above is just to name a few, but we should never assume it is the same and for an Oracle database in IaaS, no matter where it is running, we must always investigate the database AND THE infrastructure.

If a difference in performance is experienced once a database is migrated to Azure IaaS from on-premises, the following should be identified:

1. Inventory the VM SKU, not just how many vCPU or how much memory, but the EXACT SKU the database is running on.

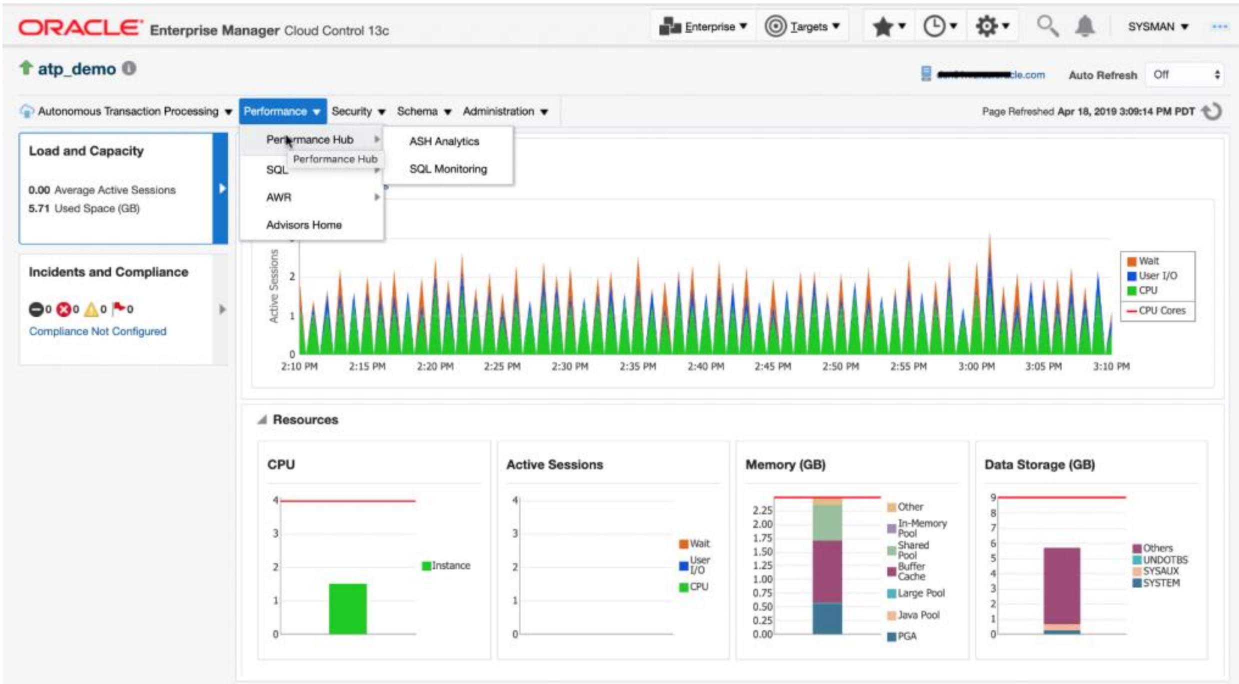
    a. If the VM is not on a #ds VM that allows for a premium SSD for OS Disk and temp storage local to the VM for the swap file, be aware of the performance implications and that steps #2 and #3 will recommend this.
2. Verify OS disk is on premium SSD storage- P6-P10 disk.
3. Verify the Linux swapfile is on the local ephemeral storage on the VM.
4. Inventory each of the attached storage- the type of storage, the size and if there is any caching turned on for each disk. Verify what datafiles/logfiles are stored on each disk.

Once the Infrastructure is verified to have all recommended practices for best performance, then it's important to inspect the database workload for optimization opportunities. The Automatic Workload Repository and other Oracle tuning products will help.

Recommendations of where to focus with Oracle optimization:

1. Inspect the AWR "Top SQL by Elapsed Time" from the previous on-premises vs. the current Azure for similar workloads.
    a. Look for outlier SQL that has degraded.
    b. Compare per single execution time in each report for same SQL for degradation
    c. Inspect top foreground and background performance for performance degradation
    d. High IO maintenance and backup jobs can put an Oracle workload into throttling. Verify that too many non-user workloads are consuming too much IO at a given time.
    e. If an upgrade was performed as part of a migration, ensure that increases in resource usage and verify no carryover of outdated parameters may have affected performance.

The Cloud Architecture and Engineering Team's Oracle Specialists commonly recommend running Oracle in Azure around 2-6 months with stability before performing a cost optimization exercise. This involves investigating and testing lower resource usage for infrastructure to save cost on the cloud. This should always be performed on a test system and should be done using the AWR report in conjunction with infrastructure information for the system. The same list as above for ***"Inspecting Performance for Oracle on Azure***" should be used but identifying where performance is more than satisfactory, using Oracle Cloud Control, (aka Enterprise Manager) and if Azure Monitor demonstrates that there is low percentage use of resources.



Once verified, then consider a lower service tier to achieve satisfactory performance, while at a better price. Recommended practice is that this is an exercise that should be performed over an extended period and to only make one change at a time, using all tools discussed to monitor any degradation in performance.