MAPR TECHNOLOGIES, INC.

REFERENCE ARCHITECTURE

OCTOBER 2017

CONVERGED DATA PLATFORM REFERENCE ARCHITECTURE FOR AZURE DEPLOYMENTS

TABLE OF CONTENTS

TABLE OF CONTENTS, CONTINUED

INTRODUCTION

This document covers the architectural aspects of deploying and operating the MapR Converged Data Platform on the Microsoft Azure platform, targeting those users who have knowledge of MapR and are getting ready to deploy on the Microsoft Azure cloud. IT and Cloud Architects who are responsible for designing and deploying MapR based big data solutions in the Microsoft Azure cloud will benefit significantly from this content. We will not get into the specifics of MapR and deployment options or the details of deploying to Microsoft Azure. We assume familiarity with MapR concepts.

We will bring in the relevant Azure services perspective for achieving the architecture quality attributes and use them as a checklist for MapR deployments to Azure. We will be selectively pointing out the MapR capabilities and features as relevant to Azure deployments. Please refer to the official MapR documentation for detailed MapR concepts and Microsoft Azure documentation for more details on Microsoft Azure.

MAPR CONVERGED DATA PLATFORM ARCHITECTURE

The MapR Converged Data Platform integrates analytics powered by Apache Drill, Apache Spark, and Apache Hadoop with real-time database capabilities, global event streaming, and scalable enterprise storage to power a new generation of big data applications. MapR Converge-X Data Fabric powers the shared services of the MapR Platform, which include high availability, unified security, multi-tenancy, disaster recovery, global namespace, resource management, automation, and real-time data access.

The architecture diagram in Figure 1 illustrates the MapR Converged Data Platform.

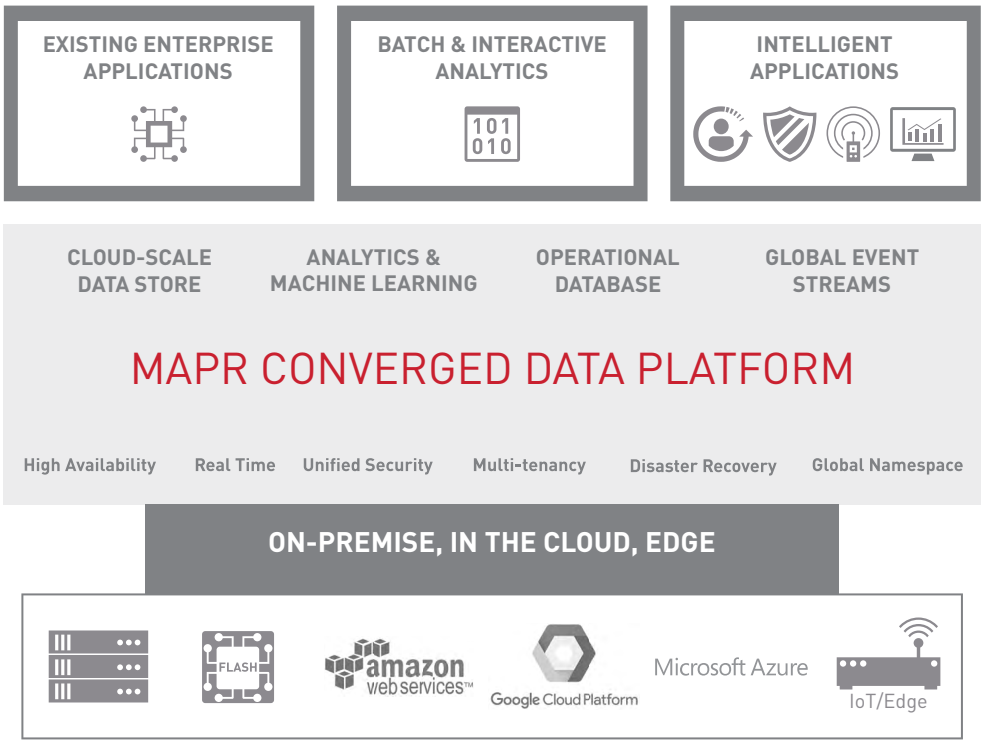

Figure 1. MapR Converged Data Platform

MAPR-XD

MapR-XD is an exabyte-scale, reliable, globally distributed data store, delivering an organization's data fabric for managing files, objects, and containers. MapR-XD supports the most stringent speed, scale, and reliability requirements within and across multiple edge, on-premises, and cloud environments.

MapR-XD supports storage pools for striping data write operations. Azure Virtual Machines (VMs) support multiple data disks. The maximum number of attached disks is determined by the VM size. Striping is a common practice for optimizing I/O on Azure VM disks. VM data disk configuration deserves a closer look, when deploying to Microsoft Azure. Please refer to the VM Disks section for the details.

MAPR-DB

MapR-DB is an enterprise-grade, high performance, multi-model NoSQL database management system that supports real-time, operational, and analytical processing. Customers use MapR-DB to manage multiple NoSQL data models, including key-value tables, wide columns, and JSON documents, which enables faster, more efficient processing of data. MapR-DB has great scale and the strong consistency needed to deploy real-time operational apps in a globally distributed environment.

MAPR-ES

MapR-ES supports processing of event-based data, including real-time data streams. MapR Streams is a publish/subscribe framework that can support the interaction of millions of producing and consuming applications at a rate of billions of events per seconds.

MAPR AZURE DEPLOYMENT FOOTPRINT

MapR instances are deployed as Azure Virtual Machines (VMs) to form the cluster. They may all depend on the same MapR-provided image on the Azure Marketplace or any supported Linux image for custom MapR deployments. Please refer to the MapR OS support matrix for details.

We will start with some important resources on Azure, related to typical MapR deployments. While a MapR deployment can be done on a single VM, starting with the Azure Marketplace image, most of the more advanced configurations will require more than one VM. We will be assuming you are making a non-trivial deployment with more than a single VM instance.

MICROSOFT AZURE NETWORKING

All VMs on Azure deployments must be associated with an Azure Virtual Network (VNet). Although deploying a single VM sounds like we will not need to deal with networking, it is important to at least understand the security implications and access to the VM.

Depending on the number of Network Interface Cards (NICs) on each VM, the VM can be in one or more subnets. We will not get into the details of the Azure Networking here. Please refer to Microsoft Azure VNet documentation for details docs.microsoft.com/en-us/azure/virtual-network.

MapR does not have any requirement regarding the number of NICs on a VM. This document assumes just one NIC per VM is used.

Network Security Groups (NSGs)

A network security group contains a list of security rules that allow or deny network traffic to resources connected to Azure VNet. You can assign NSGs to subnets or NICs. Bear in mind that all incoming and outgoing ports are open for a VM with a NIC associated with a public IP on the Azure infrastructure, and you will need to use NSGs to restrict access to the VM. Also note that the VM itself may have an OS level firewall (e.g., UFW) installed and configured, coming from the source image.

Network Bandwidth

Microsoft does not publish hard numbers on network bandwidth available for VMs; however, it is determined indirectly by the VM size selected. Please see docs.microsoft.com/en-us/azure/virtual-machines/linux/sizes.json for Azure Linux VMs.

ACCESSING THE VMS

There are multiple ways for accessing the VM deployed to the Azure platform. Let's go through them.

Public IPs

A public IP resource can be provisioned and assigned to a NIC on the VM. If this option is selected, all traffic passed through the Azure firewall will be allowed through all ports by default, and NSG rules need to be configured to restrict traffic. A lot of examples and tutorials, walking through the process of deploying a VM, are using the public IP settings, without mentioning the NSG rules. We do not recommend exposing all the nodes to the internet, without restricting traffic with NSG rules.

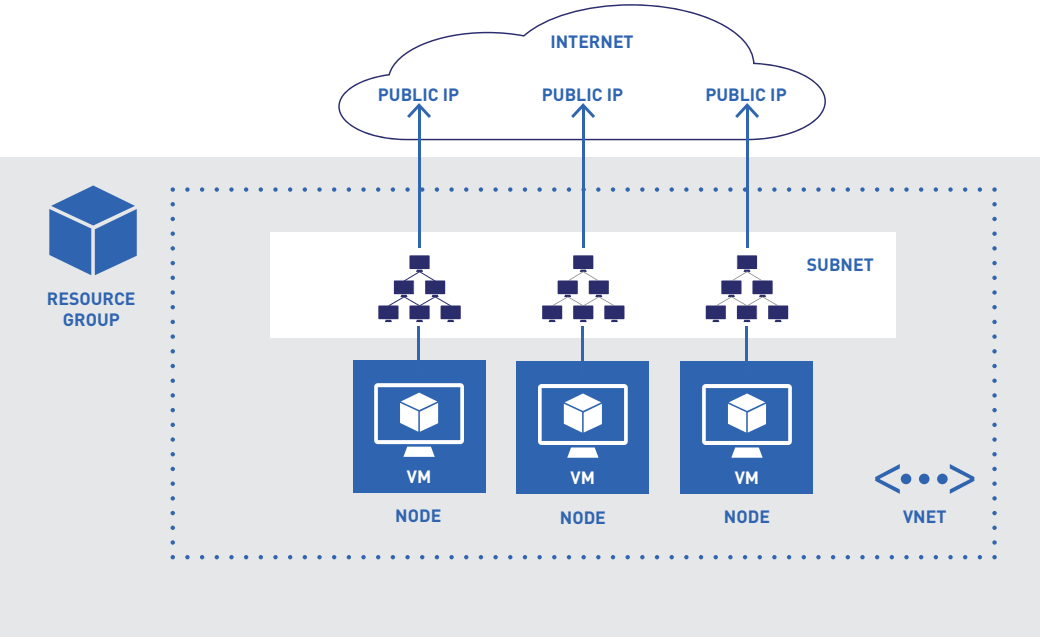

Figure 2. Exposing all nodes on the internet—NOT RECOMMENDED FOR PRODUCTION

Edge Node (Aka Jump Box in Public Azure Samples) with Public IP

With this configuration, we have one box exposed to the internet, through a public IP with allowed ports (SSH, RDP, or HTTPS). The nodes are behind this layer, and no access is provided to the nodes directly. One can notice the deficiency in this model, though, when the jump box provides access to a service (such as web) that does not have administrative access to the nodes. Although it provides client/user access to the configuration, it does not provide a clean way for accessing the nodes for administrative purposes.

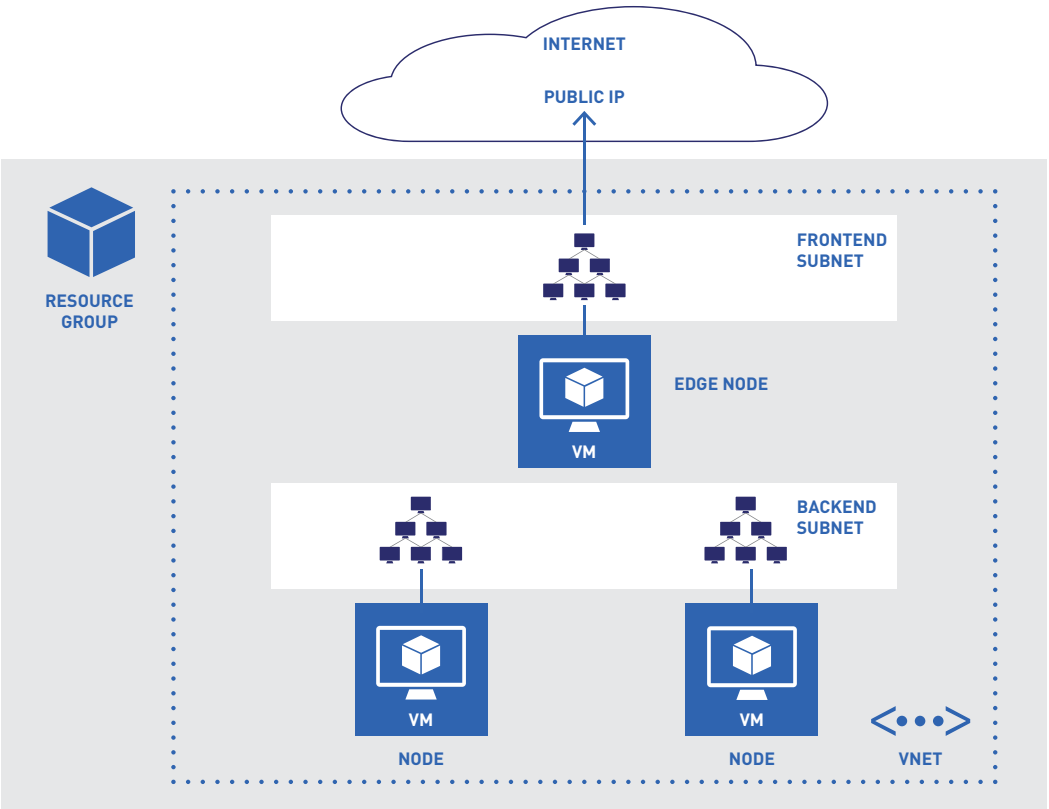

Figure 3. Exposing Only an Edge Node—NOT RECOMMENDED

Users Through the Edge Node, Administrators through VPN

The previous configuration provides a convenient way for the users of the deployment to access the services through a common protocol (e.g., SSH, HTTPS). However, it may be creating security risks, if SSH or any other remote access protocol traffic are allowed between the jump box and the backend nodes. One solution is to separate the frontend subnet from the backend subnet and provide access to all the VMs through VPN for administration purposes. Please see docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-about-vpngateways for the details. The Microsoft Azure platform provides three solutions, when connecting to an on-premises network.

Point-to-Site (VPN over SSTP)

This option can be used if the number of client machines connecting to the Azure VMs is not large, the on-site VPN device is not on the validated device list (docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-about-vpn-devices) and not compatible to be used in site-to-site (S2S) VPN

connectivity scenario, or it is simply too costly to set up a S2S connection. Please see docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-howto-point-to-site-rm-ps for the details of setting up a connection and docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-howto-point-to-site-rm-ps#a-namefaqapoint-to-site-faq for limitations.

Site-to-Site (IPsec/IKE VPN Tunnel)

This option is available if the number of client machines is not known or special VPN client software is not desired. This is the most commonly used option with the on-premises computers connecting Azure VM deployments in a secure way. Please see docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-create-site-to-site-rm-powershell for the details.

ExpressRoute

This is a dedicated private connection to extend the on-premises network into the Microsoft cloud, facilitated by a 3rd party connectivity provider. ExpressRoute connections do not go over the public internet. Please see azure.microsoft.com/en-us/services/expressroute for details.

MICROSOFT AZURE VIRTUAL MACHINES (VMS)

The Microsoft Azure platform offers several on-demand, scalable computing resources, and Azure VM is one of them. You can deploy Linux or Windows VMs. You can either use one of the Azure-provided images, through Azure Marketplace, or upload your own image.

You can deploy VMs to your Azure subscription within the limits of your subscription's quota. You can find the details on the limits and quotas here: docs.microsoft.com/en-us/azure/azure-subscription-service-limits. Always check your subscription's core limits and submit requests for increasing them, if you determine you do not have enough cores. You can check the subscription limits using PowerShell (blogs.msdn.microsoft.com/madan/2016/10/25/check-azure-resource-manager-arm-vm-core-storage-usage-using-powershell), platform independent CLI (docs.microsoft.com/en-us/azure/virtual-machines/linux/cli-manage), and Azure Portal (in the subscriptions blade, under Subscription, see usage and quotas details). Microsoft provides options for increasing quotas per subscription basis. Please contact Microsoft, when you need to modify the subscription quota.

You can control the resources the VM has, such as memory, cores, number of data disks that can be attached, local disk size, maximum cached and uncached data disk throughput, maximum NICs, and network bandwidth, indirectly, by selecting the appropriate VM family and size. Please refer to Azure Linux VM sizes at docs.microsoft.com/en-us/azure/virtual-machines/linux/sizes.

The storage option you chose for the data disks that can be attached to a VM is also an important factor in terms of management, size, and throughput.

VM Disks

VMs use virtual hard disks (VHDs) to store their operating system and data. Each Azure VM comes with one temporary disk and one operating system disk.

The temporary disk may not survive during an Azure maintenance event or redeployment of the VM. It provides short-term storage and is meant to store temporary data, such as page or swap files or application-related intermediate products. MapR does not recommend using these temporary disks as they are ephemeral and may cause data loss across system boots. Disk size and type varies by the VM family and size.

.

The operating system disk is, on the other hand, durable and backed by a page blob on Azure Storage. The size of the disk varies, depending on the operating system image the VM is based on (e.g., Linux VMs' OS disk size may change based on the selected Marketplace image or uploaded image).

You can also attach several durable data disks to the VM; each one also resides as a page blob on the associated storage account. The number of data disks that can be attached to a VM varies, depending on the VM size; the larger the VM, the more data disks it supports.

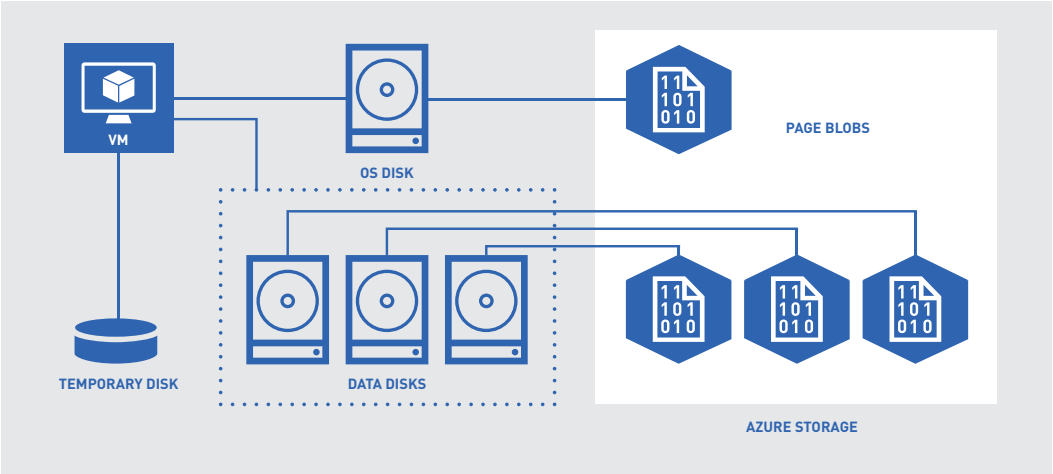

Figure 4. VM Disks

Both OS and data disks are stored in VHD format and are page blobs on the associated storage account. The data in a Microsoft Azure storage account is always replicated to ensure durability and high availability. During the storage account creation, you can select a replication option. Bear in mind that the Azure storage service is a general-purpose storage service, and only locally redundant storage (LRS) is suitable for VMs. With LRS, the Azure platform stores the blob data on three separate devices within the same data center. The Azure platform does not provide individual access to those copies, and access to a blob is transparently handled. Please see docs.microsoft.com/en-us/azure/storage/storage-redundancy for details.

The associated storage account can be a standard or premium storage account. Bear in mind that there are some limitations when premium storage is selected for storing the disks for Linux VMs, as documented at docs.microsoft.com/en-us/azure/virtual-machines/linux/azure-vm-storage-overview.

VM disks can be configured with different host cache settings. The possible values of those settings are different for OS and data disks. Please see blogs.msdn.microsoft.com/windowsazurestorage/2012/06/27/exploring-windows-azure-drives-disks-and-images for thorough coverage on how a host cache works and what those settings are. The default value of the host cache setting for OS disks is Read/Write for both storage account SKUs. The default values for data disks are None for standard SKU and Read Only for Premium Storage SKU. MapR recommends using these default settings when deploying the cluster.

Microsoft recommends setting the value to None on data disks for MapR deployments, if the solution using the disks is write-heavy, or write-only. Please see docs.microsoft.com/en-us/azure/storage/storage-premium-storage for details.

VM data disks are the primary means for achieving the required IO throughput. This is achieved in two dimensions, the storage account SKU (premium or standard) and through disk striping.

Premium storage is backed by SSDs and can deliver high-performance, low-latency disk support for intensive IO operations. Please see docs.microsoft.com/en-us/azure/storage/storage-premium-storage for details.

The other dimension to achieve higher throughput is to attach multiple data disks to the VM and stripe them. MapR-XD has direct control over the disks and can manage disks in storage pools.

Both premium and standard storage have scale and performance targets as set by the Azure platform. Please refer to docs.microsoft.com/en-us/azure/storage/storage-scalability-targets for the details.

There is a limit on the number of disks within a single storage account, due to the storage account scalability targets. As an example, if you pick the standard tier on the storage account, you can only fit 40 disks maximum before reaching the storage account imposed IOPS limit (docs.microsoft.com/en-us/azure/storage/storage-scalability-targets#unmanaged-virtual-machine-disks). Assuming you attach 8 data disks per VM and keep OS and data disks on separate storage accounts, you can only deploy 5 VM instances using the same storage account. From the IOPS perspective, this configuration will give you 8x500 IOPS (for max 4KB per operation) per VM instance on a standard storage account, assuming you are striping those disks.

Managed Disks

As the number of nodes and attached data disks increase, managing the storage accounts may become cumbersome. Microsoft introduced managed disks for overcoming this issue.

We recommend using this service as your cluster becomes larger. Azure Managed Disks require the port 8443 to be open for the outbound traffic to report the status of the disks through a VM extension to the Azure platform. Please note that MCS is using this port for inbound traffic. This does not create a conflict between the MCS and the VM managed disks extension.

VM SCALE SETS (VMSS)

The Azure platform can deploy and manage a set of **identical** VMs that you can autoscale without provisioning those instances beforehand through a compute resource called Virtual Machine Scale Sets (VMSS). Please refer to docs.microsoft.com/en-us/azure/virtual-machine-scale-sets/virtual-machine-scale-sets-overview for more details. This resource can be useful when you need to elastically scale your cluster. The VMSS can be scaled out or in, either manually or automatically, based on supported metrics (docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitoring-supported-metrics).

The VMSS require a node to be provisioned automatically through scripts, and VMs to be stateless. This is an advanced scenario, and we recommend consulting with MapR professional services.

AVAILABILITY SET (AS)

A typical data center does not have the luxury of providing redundant hardware for a service to protect against hardware failures, and we expect the cluster itself to provide high availability of the services.

An Availability Set is a logical grouping of VMs in a way that allows the platform to distribute the physical placement of those resources across the infrastructure. In the case of a planned maintenance event to the infrastructure, or a hardware or infrastructure fault, the Azure platform ensures at least one VM remains running.

The Azure AS often gets confused with AWS Availability Zones. The only similarity, however, is in the word "availability"; their underlying structure and purpose are completely different. The Azure AS is associated with one single Azure region and is used to make sure at least one VM is always up and running in the case of a failure or maintenance event.

The VMs should be grouped into separate availability sets, based on the services they are running (e.g., all of the ZooKeeper servers, serving the same site, should be on the same AS). This separation is crucial for taking advantage of the Azure platform's SLA. The Azure SLA at azure.microsoft.com/en-us/support/legal/sla/virtual-machines/v1_6 provides these service level agreements:

- For all Virtual Machines that have two or more instances deployed in the same Availability Set, we guarantee you will have Virtual Machine Connectivity to at least one instance at least 99.95% of the time.
- For any Single Instance Virtual Machine using premium storage for all Operating System Disks and Data Disks, we guarantee you will have Virtual Machine Connectivity of at least 99.9%.

Since we are now working on the Azure platform, we may want to take advantage of the AS to provide extra resiliency to our deployment.

An Availability Set achieves resiliency through fault (FD) and update domains (UDs). If Azure needs to perform a planned maintenance on the host of one of the VMs, the platform first shuts down the guest VM, then performs the update on the host.

You can set the maximum number of FDs and UDs while creating the AS.

Please refer to docs.microsoft.com/en-us/azure/virtual-machines/windows/manage-availability for details.

When working with the Availability Sets, consider:

- You can specify the AS with the location, number of update domains, and fault domains (default of 5 and 3 respectively, when deploying through APIs; if not specified, the Azure portal may use different defaults) during deployment. The fault and update domain counts can also be specified in the ARM template. Please refer to the REST API operation documentation at docs.microsoft.com/en-us/rest/api/compute/availabilitysets/availabilitysets-create for details. Also, you can notice there is no price listed for AS on the Azure pricing lists (azure.microsoft.com/en-us/pricing) i.e. it is free of charge to use AS.
- Then, you can place a VM in an AS when deploying.

The requirement for the core MapR services is to have three ZooKeeper nodes to have a quorum and at least 2 CLDB and Resource Manager instances running on the cluster. To take full advantage of the AS, we will need to configure the entire MapR cluster in a primary availability set. You also want to ensure that MapR HA services such as CLDB and RM are spread across multiple Azure FDs to avoid the negative impact resulting from hardware failure.

To explain the situation further, let's see an example. Assume you have deployed a 3 VM cluster on an AS, see the Azure Portal screen capture in Figure 5.

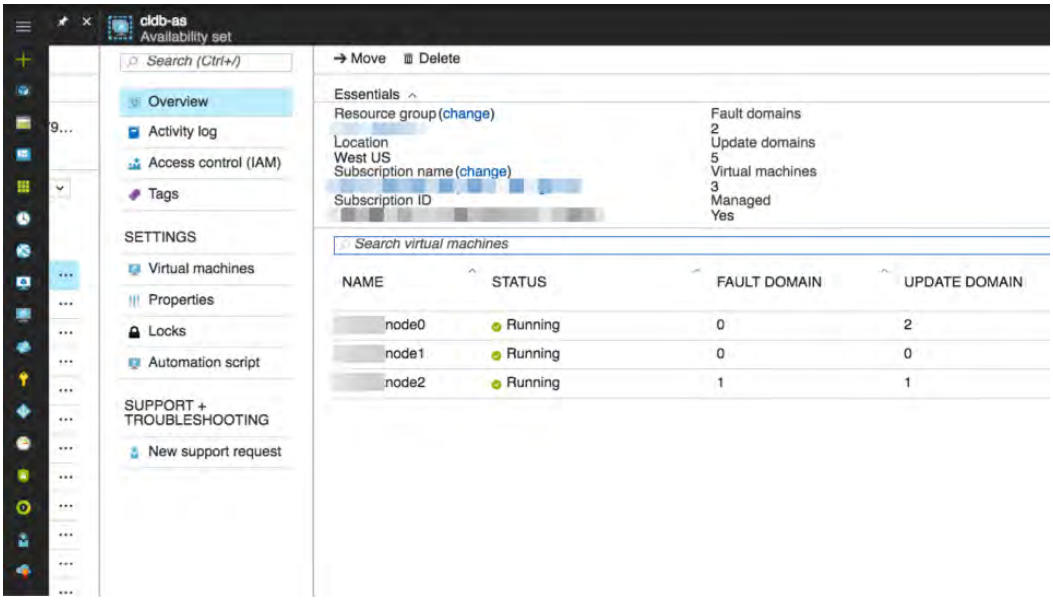

Figure 5. Azure Portal Screen Capture

As you can see, the AS has 2 fault domains and 5 update domains with 3 VMs deployed. Please note that you cannot specify which fault/update domain a VM must be placed in. Azure manages the placement of the VMs within an AS.

Also assume that you want to deploy 2 instances of the Resource Manager. As you are going through the MapR setup, you should ensure that the RM service is deployed to nodes in different fault domains to ensure that at least one instance of the RM service remains available in the event of a hardware failure. This can be done with the FD information you have from Figure 5 and the "layout" configuration page in MapR installer. In the example shown here, the RM service should be deployed to node0 and node2. However, if you do not make this conscious choice, but deploy the RM instances to node0 and node1, in the case of a hardware failure, Azure will not guarantee the RM instances to be up and running. Again, please note, this decision is on the services deployed on a VM, but not how the VMs are deployed across domains, since there is no control on that decision. Azure API allows you to query the fault domain information (i.e., which VM belongs to what FD). This information can be combined with the MapR node topology and make the cluster even more resilient. For example, you can create a topology that spans across multiple FDs, then create a volume associated with that topology. These steps will ensure that your data is distributed across multiple FDs and has less impact in the event of a hardware failure.

Our recommendation is to set the FD and UD in the AS to their maximum values, to take full advantage of FD and UD. The maximum number of FD changes per region; please see docs.microsoft.com/en-us/azure/virtual-machines/windows/manage-availability for details.

For information regarding use of the Azure API to retrieve FD information, refer to this Azure documentation: docs.microsoft.com/en-us/azure/virtual-machines/virtual-machines-instancemetadataservice-overview.

It is also possible to deploy VMs on different Availability Sets. Keeping high availability in mind, mapping the VMs in a MapR cluster to AS, and FD/UDs within them, may vary from one cluster to the other, by the cluster size and the combination of services installed. The exact topology of the MapR cluster with services distributed by AS, UD, and FD will therefore depend on your scenario. Please contact professional services for further assistance.

AZURE DEPLOYMENT OPTIONS

Customers have a few deployment options to choose from, depending on their expertise with MapR and Azure and how fast they want a MapR cluster up and running on Azure.

On the one hand, there is a MapR offering on Azure Marketplace, where MapR designed the architecture that makes sense for most of the known big data workloads out there. Customers, though, are sometimes constrained on the freedom to select the infrastructure components that give them the performance/flexibilities for their specific workloads requirements. On the other hand, customers have the absolute freedom to select the infrastructure components from Azure that meet their HA, performance, bandwidth, and security requirements, but this path would require manual deployment and Azure domain expertise—or example, the in-depth knowledge to create Azure Resource Manager (ARM) templates.

MapR on Azure Marketplace (Level: Entry)

Customers who want to get a quick start deploying MapR on Azure and experimenting with it can choose the Azure Marketplace route. MapR works closely with Azure to provide a fast path on Marketplace to deploy MapR. These customers would go through a few questions on the infrastructure design to provision a MapR cluster by simply clicking on a launch button.

The Marketplace deployment is a great way to jump start your MapR cluster on the Azure cloud—it provides you with a baseline MapR cluster that you can later expand upon as your business and data volume continue to grow.

At the writing of this document, the following VM sizes are supported:

Standard_D4s_v3, Standard_D8s_v3, Standard_D8s_v3 (These are recommended VM types)
Standard_F8s, Standard_F16s
Standard_DS3_v2, Standard_DS4_v2, Standard_DS5_v2, Standard_DS12_v2
Standard_DS13_v2, Standard_DS14_v2, Standard_DS15_v2
Standard_E4s_v3, Standard_E8s_v3, Standard_E16s_v3, Standard_E32s_v3
Standard_GS2, Standard_GS3, Standard_GS4
Standard_L4s", Standard_L8s", Standard_L16s", Standard_L32s"

The following disk types are supported, we recommend Premium_LRS:
Standard_LRS and Premium_LRS

Go to this URL and search for keyword "MapR" to get started on Marketplace: https://azuremarketplace.microsoft.com/en-us/marketplace

As you go through the wizard steps in the Marketplace, you will see most of the VM and storage types are supported as well as using new or existing VNets. You will need to make decisions on the following areas:

- **Resource Group.** A new resource group will be created for the Marketplace deployment with MapR. In the resource group, you will create VMs, network, and storage resources, subsequently. Currently, MapR deployment in an existing resource group is not supported.
- **Cluster Size.** You can determine the number of VMs in your cluster; it can be a minimum of 3 nodes.

VM Size and Disk Type. You can go with the VM sizes with fewer cores and lower memory for Proof of Concept or non-production deployments. You should opt for bigger VM sizes for any other purpose. Usually the Azure VM comes with a SSD based root partition. MapR recommends that root partition size is at least 127GB to accommodate the logs. For information on how to resize root partition, see this URL: https://docs.microsoft.com/en-us/azure/virtual-machines/windows/expand-os-disk.

- **Virtual Network (VNet).** You have a choice whether to create a new VNet or use an existing VNet in the Marketplace. Some may want to use an existing VNet for ease of network management and leveraging an existing NSG (network security group), created to comply with corporate security policies. If you select a new VNet, a NSG resource is attached to the NICs with rules allowing ports 22 and 8443 for inbound traffic, to be used by SSH and MCS, respectively.
- **Storage Type.** The MapR Platform uses MapR-XD, which allows a direct IO to storage subsystem to yield higher read/write performance, compared to HDFS that relies on the extra layer of Linux ext3 or ext4 to handle the IOs. You can choose between Standard or Premium SKUs of Azure storage, based on your performance needs. MapR-XD is composed of data disks. MapR recommends using premium Azure managed disks (embed: https://docs.microsoft.com/en-us/azure/storage/common/storage-premium-storage) for MapR-XD. You should carefully consider the disk IO throughputs by referencing this URL: https://azure.microsoft.com/en-us/pricing/details/managed-disks when you design your cluster for optimal cost/performance curve. As mentioned above, MapR recommends using default values for host cache settings.

The following is what is deployed in terms of network connectivity and VMs through the Azure Marketplace.

Please note that for security reasons, only the MCS port (8443) is open for inbound internet traffic in the Marketplace deployment. Customers are encouraged to evaluate their network requirements and modify the NSG to allow traffic to other MapR ports.

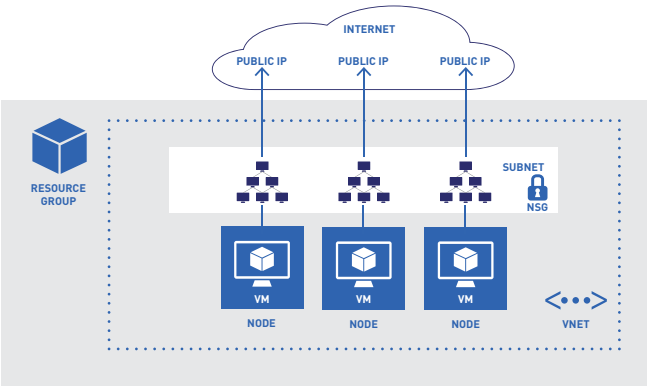

Figure 6. Deployment Footprint with Azure Marketplace—NOT RECOMMENDED FOR PRODUCTION

Manual Deployment to Microsoft Azure (Level: Intermediate to Expert)

If you want to go beyond the basic deployment through the Azure Marketplace, you will need to start making decisions for various areas, such as:

- How many subnets do I need in the Azure VNet?
- Do I need a load balancer?
- Do I need ExpressRoute for higher bandwidth between my sites and Azure?
- Which ports do I want to open on the DMZ subnet?
- Is my on-premises VPN router supported by Azure?
- What Linux OS do I want for my MapR VMs?
- How many non-MapR VMs do I need? What are their VM sizes?
- How many MapR VMs do I need? What are their VM sizes?
- What would be the number of data disks on the VM? Do I need Premium (SSD) or Standard (HDD) disk type? What is the size of each data disk to accommodate my data growth? MapR recommends Premium Managed Disks for data disks.
- What MapR services do I want to deploy?

The following deployment diagram assumes you will be deploying a multi-node MapR cluster with search and ETL clients exposed to the internet through a set of web servers, as the cluster is managed through a S2S VPN connection.

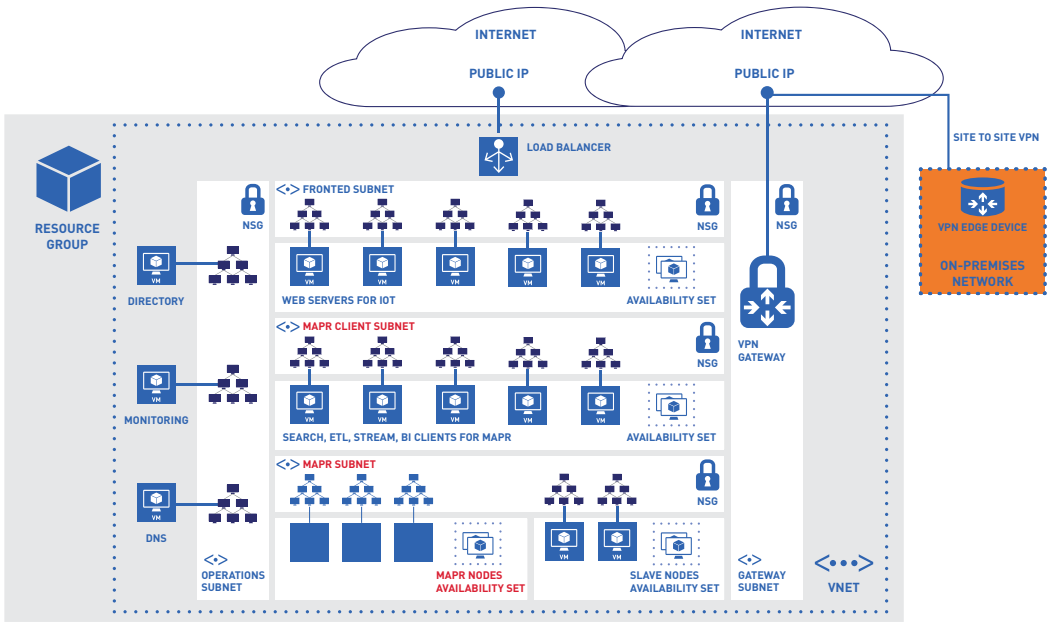

Figure 7. Example Large-Scale Deployment

You can make a larger deployment to Azure, either using the Azure Portal or through Azure Resource Manager (ARM) templates and scripting. Although using Azure Portal is more alluring, and may seem easier, we recommend using ARM templates and scripting, to be able to repeat deployments, test, and fine-tune the resources.

Please also bear in mind that the portal will eventually become one of your means for monitoring and managing your deployed resources.

Infrastructure Provisioning with Azure UI (Level: Intermediate)

Please refer to the following MapR document and blog post to provision Azure infrastructure and deploy a MapR cluster:

Deploying MapR Clusters on Azure
Manually Deploying MapR in Azure

You may want to capture the validated image after you successfully deployed MapR cluster. This captured image can serve as a base image in the future to streamline the process when adding more nodes to the MapR cluster. For information regarding how to capture an Azure VM image, follow this link: docs.microsoft.com/en-us/azure/virtual-machines/virtual-machines-linux-capture-image.

Multi-Region MapR Deployments

Customers with multiple MapR clusters located in different Azure regions might be interested in leveraging the MapR capabilities for mirroring and replication for disaster recovery (DR) purposes. In this scenario, you can establish multiple MapR clusters in different Azure geo-regions, then use the VPN gateways to link these sites for cluster mirroring and replication.

For disaster recovery, MapR volumes can be mirrored on a regular schedule from a source cluster to a target cluster. The time between mirror operations defines the recovery point objective (RPO). After the first mirroring operation, additional mirrors only copy the modified disk blocks of the mirrored volume for much faster mirroring and reduced network traffic. All mirroring network traffic can be encrypted and checksums are always calculated to ensure data consistency. In the event of a disaster at the source cluster, the mirrored volume can be promoted from read-only to read-write for use by applications at the target cluster. Promoting a volume from read-only to read/write takes only a few seconds, resulting in a very low recovery time objective (RTO). When the source cluster comes back online, the source volume for the mirror operation can become a target volume for mirroring in the opposite direction for continued disaster recovery capability. A MapR mirror guarantees that all data in the volume, whether in files, MapR-DB tables, or MapR Event Streams, is a consistent point-in-time image across all nodes in the cluster. The consistent point-in-time mirroring in the MapR Platform overcomes inconsistencies created by other backup mechanisms that copy data from one server at a time.

In addition to mirroring, MapR provides both MapR-DB tables and MapR Streams replication. As data is added to MapR tables and event streams, the replicated table or stream on the target cluster is updated within a few seconds. This ensures a very low RPO, measured in seconds. The table or stream on the target cluster is always read/write enabled, so RTO is simply the time required to redirect clients to the table or stream on the target cluster.

The figure on the next page describes a multi-site MapR deployment across Azure regions. Please refer to mapr.com/resources/disaster-recovery for information about the DR capabilities of the MapR Platform.

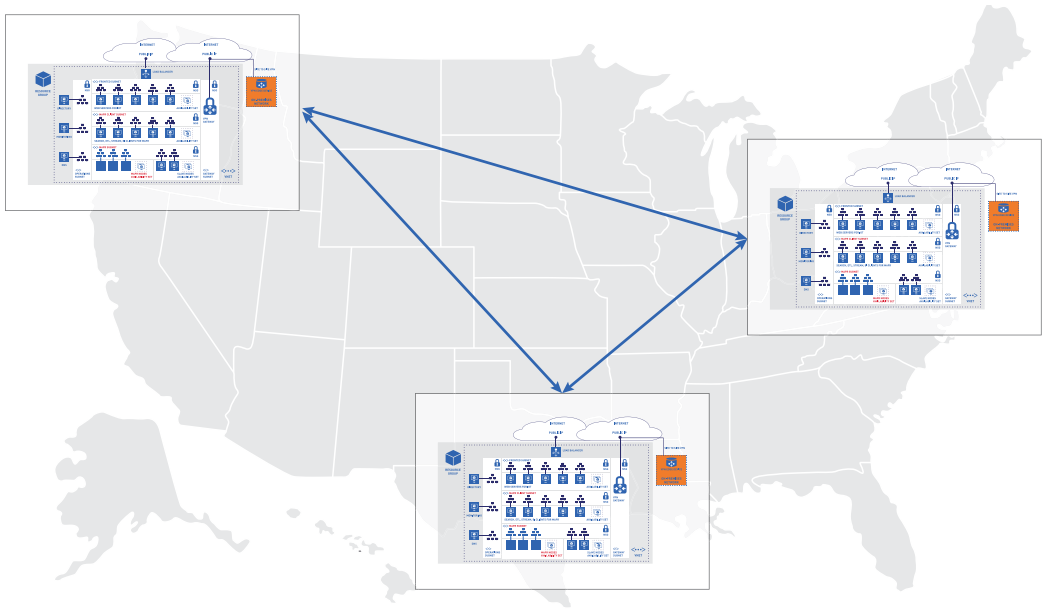

Figure 8. Multi-Region Replication of MapR Clusters on Azure

You may also consider a multi-region deployment for DR and HA purposes, accessible through one single DNS entry. The Microsoft Azure platform provides a feature called Azure Traffic Manager that enables such a scenario. This is an advanced case, and we recommend working with MapR professional services in this case.

PREDOMINANT ARCHITECTURAL QUALITY ATTRIBUTES

Every robust system needs to follow well-established patterns and best practices, irrespective of the functional requirements for their intended services. Some of those are also known as non-functional requirements. We will be using a subset of those requirements, as MapR deployments and Azure platform services converge for a checklist on the areas we have covered above.

PERFORMANCE

Performance of the deployed cluster depends on the size of the cluster and the resources available for each node. One important point is about the maximum IOPS on the VM. The maximum IOPS attained on a VM is dependent on the maximum number of data disks that can be attached to the VM as determined by the VM size, their striping configuration, and the underlying SKU (Standard or Premium) of the Azure Storage account. Please see the section, "Microsoft Azure Virtual Machines (VMs)" on page 8, above, for more details.

SCALABILITY

The MapR Converged Data Platform provides high availability for the Hadoop components in the stack. MapR clusters don't use NameNodes and provide stateful high availability for the MapReduce JobTracker and Direct Access NFS; the default setup works out of the box with no special configuration required.

AVAILABILITY

MapR clusters support Azure region availability. The MapR clustering architecture ensures the high availability of MapR services to its users right out of the box. If you consider high availability through multi-region deployments, please contact MapR Professional Services.

Disaster recovery (DR) is also a very important aspect that should not be omitted. The Azure platform provides services for backing up, protecting, and replicating VMs. We do not recommend the use of those services in the context of MapR deployments. Please consider using MapR service-level disaster recovery mechanisms for ensuring the consistency of your clusters.

RELIABILITY

The VM data disks are stored as blobs on Azure Storage. Azure Storage keeps at least three copies of the data on the disk with LRS setting. MapR services also manage the data when configured for replication to provide reliable access to it, and you can achieve global reliability through multi-region deployments.

SECURITY

Places for securing the deployment are available at multiple levels. From the Azure platform perspective, at minimum, customers should consider using NSGs to control the access to the TCP ports and routing rules and manage access to the subscription itself through role-based access control.

Consider using the activity logs on the Azure platform (docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitoring-overview-activity-logs) as well as Network Watcher (docs.microsoft.com/en-us/azure/network-watcher) for monitoring the network activity.

Microsoft has published many security best practices documents at docs.microsoft.com/en-us/azure/security/security-best-practices-and-patterns and specifically for networks at docs.microsoft.com/en-us/azure/best-practices-network-security. Please refer to MapR documentation for securing the cluster at the MapR services level.

Azure supports encrypting the data disks attached to the VMs. Please see docs.microsoft.com/en-us/azure/security/azure-security-disk-encryption for details.

To get started on MapR level security, please refer to the MapR security guide here: maprdocs.mapr.com/home/SecurityGuide/Getting-Started-MapR-Security.html?hl=security.

SUPPORTABILITY AND MANAGEABILITY

Consider using a combination of MapR Control System (MCS), MapR monitoring, and Azure activity logs to manage and diagnose the deployments. Azure Storage also exposes a very detailed analytics functionality to drill down into the storage level; please see docs.microsoft.com/en-us/azure/storage/storage-analytics for details.

Microsoft Azure Network Watcher is also a great resource for monitoring the Azure network; please refer to docs.microsoft.com/en-us/azure/network-watcher/ for details.

MAINTAINABILITY

Maintenance on the MapR level should follow the published and established practices. MapR clusters are deployed to Azure Infrastructure as a Service (IaaS) resources, and updating the underlying operating system is not different than the standard on-premises practices. However, deploying to a cloud environment comes with its own perks to minimize the downtime. We encourage users to utilize various scripting methods in their deployments for repeatability over using the Azure portal. Please refer to Azure PowerShell (docs.microsoft.com/en-us/powershell/azure/overview?view=azurermps-3.7.0) or Azure platform independent Command Line Interface (CLI) (docs.microsoft.com/en-us/cli/azure/overview).

REFERENCES

MAPR CONVERGED DATA PLATFORM

Platform Overview

Deploy MapR on Azure

MapR Disaster Recovery

MapR Converge Community

MapR Blog

MICROSOFT AZURE

MapR Documentations

Azure Homepage

Azure Resource Manager

Azure IaaS

Azure Virtual Machines

Azure Virtual Network

Azure Premium Storage

Azure Availability Sets

Azure ExpressRoute

Azure Subscription and Service Limits, Quotas, and Constraints

Azure Services

Azure Network Security Best Practices

Azure Resource Manager Template

Azure Managed Disks