

DBMS Deployment Guide for SAP on Microsoft Azure Virtual Machine Services

Microsoft Corporation

Version: 3.10

Date of last change: 10/08/2015

**Abstract**

This guide is part of the documentation on implementing and deploying the SAP software on Microsoft Azure. Before reading this guide, please read the document titled ‘*SAP NetWeaver on Microsoft Azure Virtual Machine Services – Planning and Implementation Guide*’. This document covers the deployment of various Relational Database Management Systems (RDBMS) and related products in combination with SAP on Microsoft Azure Virtual Machines (VMs) using the Azure **I**nfrastructure **a**s **a** **S**ervice (IaaS) capabilities. The first release of the guide focused on Microsoft SQL Server. The second release of the guide adds details for deployments of SAP on top of SAP ASE and Oracle RDBMS in conjunction with SAP deployments on Azure. The current release includes additions for SAP MaxDB, LiveCache and Content Server.

The paper complements the SAP Installation Documentation and SAP Notes which represent the primary resources for installations and deployments of SAP software on given platforms.

**Copyright information**

This document is provided "as-is". Information and views expressed in this document, including URL and other Internet website references, may change without notice.

Some examples depicted herein are provided for illustration only and are fictitious. No real association or connection is intended or should be inferred.

This document does not provide you with any legal rights to any intellectual property in any Microsoft product. You may copy and use this document for your internal, reference purposes.

Microsoft, Active Directory, Hyper-V, SQL Server, Windows PowerShell, Windows, Microsoft Azure and Windows Server are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. All other trademarks are property of their respective owners.

© 2014 Microsoft Corporation. All rights reserved.

[1 General considerations 6](#_Toc432101761)

[1.1 Definitions upfront 6](#_Toc432101762)

[1.2 Resources 7](#_Toc432101763)

[2 Structure of an RDBMS Deployment 9](#_Toc432101764)

[2.1 Caching for VMs and VHDs 10](#_Toc432101765)

[2.2 Striping and Storage Spaces 11](#_Toc432101766)

[2.3 Microsoft Azure Storage 12](#_Toc432101767)

[2.3.1 Mapping VHDs into Azure Virtual Machine Service Storage Accounts 14](#_Toc432101768)

[2.3.2 Moving deployed DBMS VMs from Azure Standard Storage to Azure Premium Storage 15](#_Toc432101769)

[2.4 Deployment of VMs for SAP in Azure 16](#_Toc432101770)

[3 High Availability and Disaster Recovery with Azure VMs 18](#_Toc432101771)

[3.1 VMs deployed on Azure Nodes 18](#_Toc432101772)

[3.1.1 IP Addresses 20](#_Toc432101773)

[4 Deployment of SAP related and required Host Monitoring capabilities 21](#_Toc432101774)

[5 Specifics to SQL Server RDBMS 22](#_Toc432101775)

[5.1 SQL Server IaaS 22](#_Toc432101776)

[5.2 SQL Server configuration guidelines for SAP related SQL Server installations in Azure VMs 23](#_Toc432101777)

[5.2.1 Recommendations on VM/VHD structure for SAP related SQL Server deployments 23](#_Toc432101778)

[5.2.2 Formatting the VHDs 24](#_Toc432101779)

[5.2.3 Impact of database compression 24](#_Toc432101780)

[5.3 SQL Server 2014 – Storing Database Files directly on Azure Blog Storage 25](#_Toc432101781)

[5.4 SQL Server 2014 Buffer Pool Extension 26](#_Toc432101782)

[5.5 Backup/Recovery considerations for SQL Server 26](#_Toc432101783)

[5.5.1 SQL Server 2012 SP1 CU4 and later 26](#_Toc432101784)

[5.5.2 SQL Server 2012 SP1 CU3 and earlier releases 28](#_Toc432101785)

[5.5.3 Other possibilities to backup SQL Server databases 29](#_Toc432101786)

[5.5.4 Performance considerations for backups/restores 30](#_Toc432101787)

[5.5.5 Managing Backup BLOBs 30](#_Toc432101788)

[5.5.6 Azure VM backup 31](#_Toc432101789)

[5.6 Using a SQL Server images out of the Microsoft Azure Gallery 31](#_Toc432101790)

[5.6.1 Changing the SQL Server Collation of a Microsoft Windows/SQL Server VM 31](#_Toc432101791)

[5.7 SQL Server High-Availability for SAP in Azure 32](#_Toc432101792)

[5.7.1 SQL Server Log Shipping 32](#_Toc432101793)

[5.7.2 Database Mirroring 33](#_Toc432101794)

[5.7.3 AlwaysOn 33](#_Toc432101795)

[5.7.4 Summary on SQL Server High Availability in Azure 34](#_Toc432101796)

[5.8 General SQL Server for SAP on Azure Summary 35](#_Toc432101797)

[6 Specifics to SAP Adaptive Server (SAP ASE) RDBMS 37](#_Toc432101798)

[6.1 SAP ASE Version Support 37](#_Toc432101799)

[6.2 SAP ASE Configuration Guidelines for SAP related SAP ASE Installations in Azure VMs 37](#_Toc432101800)

[6.2.1 Structure of the SAP ASE Deployment 37](#_Toc432101801)

[6.2.2 Impact of Database Compression 38](#_Toc432101802)

[6.2.3 Using DBACockpit to monitor Database Instances 39](#_Toc432101803)

[6.2.4 Backup/Recovery Considerations for SAP ASE 40](#_Toc432101804)

[6.2.5 Performance Considerations for Backups/Restores 41](#_Toc432101805)

[6.3 Disaster Recovery with Azure VMs 42](#_Toc432101806)

[6.3.1 Data Replication with SAP Sybase Replication Server 42](#_Toc432101807)

[7 Specifics to Oracle Database 43](#_Toc432101808)

[7.1 Oracle Version Support 43](#_Toc432101809)

[7.2 Oracle Configuration Guidelines for SAP Installations in Azure VMs 43](#_Toc432101810)

[7.2.1 Storage configuration 43](#_Toc432101811)

[7.2.2 Backup / Restore 44](#_Toc432101812)

[7.2.3 High Availability 44](#_Toc432101813)

[7.2.4 Other 44](#_Toc432101814)

[8 Specifics for the SAP MaxDB Database 45](#_Toc432101815)

[8.1 SAP MaxDB Version Support 45](#_Toc432101816)

[8.2 Supported Microsoft Windows Versions and Azure VM types for SAP MaxDB DBMS 45](#_Toc432101817)

[8.3 Available SAP MaxDB Documentation 45](#_Toc432101818)

[8.4 SAP MaxDB Configuration Guidelines for SAP Installations in Azure VMs 45](#_Toc432101819)

[8.4.1 Storage configuration 45](#_Toc432101820)

[8.4.2 Backup and Restore 46](#_Toc432101821)

[8.4.3 Performance Considerations for Backup and Restore 47](#_Toc432101822)

[8.4.4 Other 47](#_Toc432101823)

[9 Specifics for SAP liveCache 49](#_Toc432101824)

[9.1 SAP liveCache Version Support 49](#_Toc432101825)

[9.2 Supported Microsoft Windows Versions and Azure VM types for SAP liveCache DBMS 49](#_Toc432101826)

[9.3 SAP liveCache Configuration Guidelines for SAP Installations in Azure VMs 49](#_Toc432101827)

[9.3.1 Recommended Azure VM Types 49](#_Toc432101828)

[9.3.2 Storage Configuration 49](#_Toc432101829)

[9.3.3 Dedicated Azure VM for liveCache 50](#_Toc432101830)

[9.3.4 Backup and Restore 50](#_Toc432101831)

[9.3.5 Other 50](#_Toc432101832)

[10 Specifics for the SAP Content Server 51](#_Toc432101833)

[10.1 SAP Content Server Version Support 51](#_Toc432101834)

[10.2 Supported Microsoft Windows and Azure VM types for SAP Content Server 51](#_Toc432101835)

[10.3 SAP Content Server Configuration Guidelines for SAP Installations in Azure VMs 51](#_Toc432101836)

[10.3.1 Storage Configuration 51](#_Toc432101837)

[10.3.2 SAP Content Server Location 52](#_Toc432101838)

[10.3.3 SAP Cache Server Location 52](#_Toc432101839)

[10.3.4 Backup / Restore 53](#_Toc432101840)

[10.3.5 Other 54](#_Toc432101841)

## 

# General considerations

In this chapter, considerations of running SAP related DBMS systems in Azure VMs are introduced. There are few references to specific DBMS systems in this chapter. Instead the specific DBMS systems are handled within this paper, after this chapter.

## Definitions upfront

Throughout the document we will use the following terms:

* IaaS: Infrastructure as a Service.
* PaaS: Platform as a Service.
* SaaS: Software as a Service.
* SAP Component: an individual SAP application such as ECC, BW, Solution Manager or EP.  SAP components can be based on traditional ABAP or Java technologies or a non-NetWeaver based application such as Business Objects.
* SAP Environment: one or more SAP components logically grouped to perform a business function such as Development, QAS, Training, DR or Production.
* SAP Landscape: This refers to the entire SAP assets in a customer’s IT landscape. The SAP landscape includes all production and non-production environments.
* SAP System: The combination of DBMS layer and application layer of e.g. an SAP ERP development system, SAP BW test system, SAP CRM production system, etc. In Azure deployments it is not supported to divide these two layers between on-premises and Azure. This means an SAP system is either deployed on-premises or it is deployed in Azure. However, you can deploy the different systems of an SAP landscape into either Azure or on-premises. For example, you could deploy the SAP CRM development and test systems in Azure but the SAP CRM production system on-premises.
* Cloud-Only deployment: A deployment where the Azure subscription is not connected via a site-to-site or ExpressRoute connection to the on-premises network infrastructure. In common Azure documentation these kinds of deployments are also described as ‘Cloud-Only’ deployments. Virtual Machines deployed with this method are accessed through the internet and public internet endpoints assigned to the VMs in Azure. The on-premises Active Directory (AD) and DNS is not extended to Azure in these types of deployments. Hence the VMs are not part of the on-premises Active Directory. **Note: Cloud-Only deployments in this document is defined as complete SAP landscapes are running exclusively in Azure without extension of Active Directory or name resolution from on-premises into public cloud. Cloud-Only configurations are not supported for production SAP systems or configurations where SAP STMS or other on-premises resources need to be used between SAP systems hosted on Azure and resources residing on-premises.**
* Cross-Premises: Describes a scenario where VMs are deployed to an Azure subscription that has site-to-site, multi-site or ExpressRoute connectivity between the on-premises datacenter(s) and Azure. In common Azure documentation, these kinds of deployments are also described as Cross-Premises scenarios. The reason for the connection is to extend on-premises domains, on-premises Active Directory and on-premises DNS into Azure. The on-premises landscape is extended to the Azure assets of the subscription. Having this extension, the VMs can be part of the on-premises domain. Domain users of the on-premises domain can access the servers and can run services on those VMs (like DBMS services). Communication and name resolution between VMs deployed on-premises and Azure deployed VMs is possible. This is the scenario we expect most SAP assets to be deployed in. See more information here: <http://msdn.microsoft.com/en-us/library/azure/jj156075.aspx>. N**ote: Cross-Premises deployments of SAP systems where Azure Virtual Machines running SAP systems are members of an on-premises domain are supported for production SAP systems. Cross-Premises configurations are supported for deploying parts or complete SAP landscapes into Azure. Even running the complete SAP landscape in Azure requires having those VMs being part of on-premises domain and ADS. In former versions of the documentation we talked about Hybrid-IT scenarios, where the term ‘Hybrid’ is rooted in the fact that there is a cross-premises connectivity between on-premises and Azure. Plus, the fact that the VMs in Azure are part of the on-premises Active Directory.**

Some Microsoft documentation describes Cross-Premises scenarios a bit differently, especially for DBMS HA configurations. In the case of the SAP related documents, the Cross-Premises scenario just boils down to having a site-to-site or private (ExpressRoute) connectivity and the fact that the SAP landscape is distributed between on-premises and Azure.

## Resources

The following guides are available for the topic of SAP deployments on Azure:

* SAP NetWeaver on Microsoft Azure Virtual Machine Services – Planning and Implementation Guide
* SAP NetWeaver on Microsoft Azure Virtual Machine Services – Deployment Guide
* DBMS Deployment Guide for SAP on Microsoft Azure Virtual Machine Services – this document
* SAP NetWeaver: Building a Microsoft Azure–based Disaster Recovery Solution
* Clustering SAP ASCS Instance using Windows Server Failover Cluster on Azure with SIOS DataKeeper

The guides can be downloaded under the section SAP here: <http://go.microsoft.com/fwlink/p/?LinkId=397566>

The following SAP Notes are related to the topic of SAP on Azure:

|  |  |
| --- | --- |
| Note number | Title |
| [1928533](http://service.sap.com/sap/support/notes/1928533) | SAP Applications on Azure: Supported Products and Sizing |
| [2015553](http://service.sap.com/sap/support/notes/2015553) | SAP on Microsoft Azure: Support Prerequisites |
| [1999351](http://service.sap.com/sap/support/notes/1999351) | Enhanced Azure Monitoring for SAP |
| [2178632](http://service.sap.com/sap/support/notes/2178632) | Key Monitoring Metrics for SAP on Microsoft Azure |
| [1409604](http://service.sap.com/sap/support/notes/1409604) | Virtualization on Windows: Enhanced Monitoring |
| [2039619](http://service.sap.com/sap/support/notes/2039619) | SAP Applications on Azure using Oracle Database: Supported Products and Versions |

You should have a working knowledge about the Microsoft Azure Architecture and how Microsoft Azure Virtual Machines are deployed and operated. You can find more information here: <http://azure.microsoft.com/en-us/documentation/>

Please note we are **not** discussing Microsoft Azure Platform as a Service (PaaS) offerings of the Microsoft Azure Platform. The discussion in this paper is about running a database management system (DBMS) in Microsoft Azure Virtual Machines (IaaS) just as you would run the DBMS in your on-premises solution. Database capabilities and functionalities between these two offers are very different and should not be mixed up with each other. See also: <http://www.windowsazure.com/en-us/services/data-management/>

Since we are discussing IaaS, in general the Windows and DBMS installation and configuration are essentially the same as any virtual machine or bare metal machine you would install on-premises. However, there are some architecture and system management implementation decisions which will be different when utilizing IaaS. The purpose of this document is to explain the specific architectural and system management differences that you must be prepared for when using IaaS.

In general, the overall areas of difference that this paper will discuss are:

* Planning the proper VM/VHD layout of SAP systems to ensure you have the proper data file layout and can achieve enough IOPS for your workload.
* Networking considerations when using IaaS.
* Specific database features to use in order to optimize the database layout.
* Backup and restore considerations in IaaS.
* Utilizing different types of images for deployment.
* High Availability in Azure IaaS.

# Structure of an RDBMS Deployment

In order follow this chapter, it is necessary to understand what was presented in Chapter 3 of the document: ‘*SAP NetWeaver on Microsoft Azure Virtual Machine Services – Planning and Implementation Guide*’. Knowledge on the different VM-Series and their differences and differences of Azure Standard and Premium Storage should be understood and known before reading this chapter.

Until March 2015, Azure VHDs which contain an operating system were limited to 127 GB in size. This limitation got lifted in March 2015 (for more information check <http://azure.microsoft.com/blog/2015/03/25/azure-vm-os-drive-limit-octupled/> ). From there on VHDs contain the operating system can have the same size as any other VHD. Nevertheless, we still prefer a structure of deployment where the operating system, DBMS and eventual SAP binaries are separate from the database files. Therefore, we expect SAP systems running in Azure Virtual Machines will have the base VM (or VHD) installed with the operating system, database management system executables and SAP executables. The DBMS data and log files will be stored in Azure Storage (Standard or Premium Storage) in separate VHD files and attached as logical disks to the original Azure operating system image VM.

Dependent on leveraging Azure Standard or Premium Storage (by using the DS-series VMs) there are other quotas in Azure which are documented here: <http://msdn.microsoft.com/library/azure/dn197896.aspx>. When planning your Azure VHDs, you’ll need to find the best balance of the quotas for the following:

* The number of data files.
* The number of VHDs which contain the files.
* The IOPS quotas of a single VHD.
* The data throughput per VHD.
* The number of additional VHDs possible per VM size.
* The overall storage throughput a VM can provide.

Azure will enforce an IOPS quota per VHD drive. These quotas are different for VHDs hosted on Azure Standard Storage and Premium Storage. I/O latencies will also be very different between the two storage types with Premium Storage delivering factors better I/O latencies. Each of the different VM types have a limited number of VHDs that you are able to attach. Another restriction is that only certain VM types can leverage Azure Premium Storage. This means the decision for a certain VM type might not only be driven by the CPU and memory requirements, but also by the IOPS, latency and disk throughput requirements that usually are scaled with the number of VHDs or the type of Premium Storage disks. Especially with Premium Storage the size of a VHD also might be dictated by the number of IOPS and throughput that needs to be achieved by each VHD.

The fact that the overall IOPS rate, the number of VHDs mounted, and the size of the VM are all tied together, might cause an Azure configuration of an SAP system to be different than its on-premises deployment. The IOPS limits per LUN are usually configurable in on-premises deployments. Whereas with Azure Storage those limits are fixed or as in Premium Storage dependent on the disk type. So with on-premises deployments we see customer configurations of database servers which are using many different volumes for special executables like SAP and the DBMS or special volumes for temporary databases or table spaces. When such an on-premises system is moved to Azure it might lead to a waste of potential IOPS bandwidth by wasting a VHD for executables or databases which do not perform any or a lot of IOPS. Therefore, in Azure VMs we recommend that the DBMS and SAP executables be installed in the base VM (C:\ drive) if possible.

The placement of the database files and log files and the type of Azure Storage used, should be defined by IOPS, latency and throughput requirements. In order to have enough IOPS for the transaction log, you might be forced to leverage multiple VHDs for the transaction log file or use a larger Premium Storage disk. In such a case one would simply build a stripe set or Windows Storage Pool over the VHDs which will contain the transaction log.

Drive D:\ in an Azure VM is a non-persisted drive which is backed by some local disks on the Azure compute node. Because it is non-persisted, this means that any changes made to the content on the D:\ drive is lost when the VM is rebooted. By any changes, we mean files saved, directories created, applications installed, etc. Dependent on the Azure VM-series, the local disks on the compute node show different performance which can be categorized like:

* A0-A7: Very limited performance. Not usable for anything beyond windows page file
* A8-A11: Very good performance characteristics with some ten thousand IOPS and >1GB/sec throughput
* D-Series: Very good performance characteristics with some ten thousand IOPS and >1GB/sec throughput
* DS-Series: Very good performance characteristics with some ten thousand IOPS and >1GB/sec throughput
* G-Series: Very good performance characteristics with some ten thousand IOPS and >1GB/sec throughput

Statements above are applying to the VM types that are certified with SAP. The VM-series with excellent IOPS and throughput qualify for leverage by some DBMS features, like tempdb or temporary table space.

## Caching for VMs and VHDs

When we create those disks/VHDs through the portal or when we mount uploaded VHDs to VMs, we can choose whether the I/O traffic between the VM and those VHDs located in Azure storage are cached. Azure Standard and Premium Storage use two different technologies for this type of cache. In both cases the cache itself would be disk backed on the same drives used by the D:\ drive of the VM.

For Azure Standard Storage the possible cache types are:

* No caching
* Read caching
* Read and Write caching

In order to get consistent and deterministic performance, you should set the caching on Azure Standard Storage for all VHDs containing **DBMS related data files, log files and table space to ‘NONE’.** The caching of the VM can remain with the default.

For Azure Premium Storage the following caching options exist:

* No caching
* Read caching

Recommendation for Azure Premium Storage is to leverage **read caching for data files** of the SAP database and **chose no cache for the VHD(s) the log file(s)** are located on.

## Striping and Storage Spaces

As already stated above, you will need to balance the number of IOPS needed for the database files across the number of VHDs you can configure and the maximum IOPS an Azure VM will provide per VHD or Premium Storage disk type. Easiest way to deal with the IOPS load over VHDs is to build stripe sets or Windows Storage Pools over the different VHDs. Then place a number of data files of the SAP DBMS on the LUNS carved out of the stripe or storage pools. Dependent on the requirements you might want to consider the usage of Premium Storage as well since two of the three different Premium Storage disks provide higher IOPS quota than VHDs based on Standard Storage. Besides the significant better I/O latency provided by Azure Premium Storage.

Same applies to the transaction log of the different DBMS systems. With a lot of them just adding more Tlog files does not help since the DBMS systems write into one of the files at a time only. If higher IOPS rates are needed than a single Standard Storage based VHD can deliver, you can stripe or use Windows Storage Pools over multiple Standard Storage VHDs or you can use a larger Premium Storage disk type that beyond higher IOPS rates also delivers factors lower latency for the write I/Os into the transaction log.

Situations experienced in Azure deployments which would favor using Windows Storage Spaces (<http://technet.microsoft.com/en-us/library/hh831739.aspx> ) or Windows Striping can be listed like:

* Transaction Log/Redo Log require more IOPS than Azure provides for a single VHD. As mentioned above this can be solved by building a LUN over multiple VHDs using Storage Spaces or Windows Striping.
* Uneven I/O workload distribution over the different data files of the SAP database. In such cases one can experience one data file hitting the quota rather often. Whereas other data files are not even getting close to the IOPS quota of a single VHD. In such cases the easiest solution is to build one LUN over multiple VHDs using Storage Spaces or Windows Striping.
* You don’t know what the exact I/O workload per data file is and only roughly now what the overall IOPS workload against the DBMS is. Easiest to do is to build one LUN with the help of Storage Space or Windows striping that with the sum of quotas of the multiple VHDs fulfill the known IOPS rate.

Usage of Windows Server 2012 or higher Storage Spaces is preferable since it is more efficient that Windows Striping of earlier Windows versions.

Please be aware that you might need to create the Windows Storage Pools and Storage Spaces by PowerShell commands when using Windows Server 2012 as Operating System. The PowerShell commands and sequence are documented in the Appendix of the whitepaper which can be downloaded here:   
<https://msdn.microsoft.com/en-us/library/azure/dn133149.aspx>

Considerations for leveraging VM-series that is able to work with Azure Premium Storage usually are:

* Demands for I/O latencies that are close to what SAN/NAS devices deliver.
* Demand for factors better I/O latency than Azure Standard Storage can deliver.
* Higher IOPS per VM that could be achieved with multiple Standard Storage VHDs against a certain VM type.

Since the underlying Azure Storage replicates each VHD to at least three storage nodes, simple RAID 0 striping can be used. There is no need to implement RAID5 or RAID1.

## Microsoft Azure Storage

Microsoft Azure Storage will store the base VM (with OS) and VHDs or BLOBs to at least 3 separate storage nodes. When creating a storage account, there is a choice of protection as shown here:

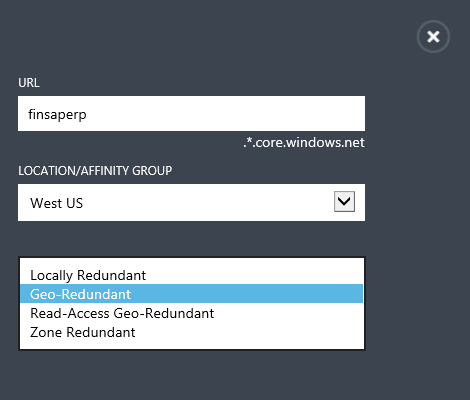


Figure Geo-Replication enabled for Azure Storage account

Azure Storage Local Replication provides levels of protection against data loss due to infrastructure failure that few customers could afford to deploy. As shown above there are 3 different option with a fourth being a variation of one of the first three. Looking closer at them we can distinguish:

* **Local Redundant Storage (LRS)**: In this case there are 3 replicas of the data within the same Azure datacenter of an Azure region. The copies will be in different Fault and Upgrade Domains (for concepts see chapter 3 in the guide: ‘*SAP NetWeaver on Microsoft Azure Virtual Machine Services – Planning and Implementation Guide’*). In case of a replica of the data going out of service due to a storage node failure or disk failure, a new replica is generated automatically.
* **Zone Redundant Storage (ZRS):** In this case the 3 replicas of the data remain in the same Azure Region. As explained in chapter 3 of ‘*SAP NetWeaver on Microsoft Azure Virtual Machine Services – Planning and Implementation Guide’* an Azure region can be a number of datacenter in close proximity. In the case of LRS the replicas would be distributed over the different datacenters that make one Azure region.
* **Geo Redundant Storage (GRS)**: In this case there is an asynchronous replication that will feed an additional 3 replicas of the data in another Azure Region which in most of the cases in the same geographical region (like North Europe and West Europe). This will result in 3 additional replicas, so that there are 6 replicas in sum. A variation of this is an addition where the data in the geo replicated Azure region can be used for read purposes (**Read-Access Geo-Redundant**).

Please note that as of July 2015, GRS is not offered for Azure Premium Storage. More information can be found here: <http://azure.microsoft.com/en-us/documentation/articles/storage-redundancy/>

**For DBMS deployments, the usage of Geo Redundant Storage is not recommended**. Azure Storage Geo-Replication is asynchronous. Replication of individual VHDs mounted to a single VM are not synchronized in lock step. Therefore, it is not suitable to replicate DBMS files that are distributed over different VHDs or deployed against a stripe set or storage pool based on multiple VHDs. DBMS software requires that the persistent disk storage is precisely synchronized across different LUNs and underlying disks/VHDs/spindles. DBMS software uses various mechanisms to sequence IO write activities and a DBMS will report that the disk storage targeted by the replication is corrupted if these vary even by a few milliseconds. Hence if one really wants a database configuration with a database stretched across multiple VHDs geo-replicated, such a replication needs to be performed with database means and functionality. One should not rely on Azure Storage Geo-Replication to perform this job.

The problem is simplest to explain with an example system. Let’s assume you have an SAP system uploaded into Azure which has 8 VHDs containing data files of the DBMS plus one VHD containing the transaction log file. Each one of these 9 VHDs will have data written to them in a consistent method according to the DBMS, whether the data is being written to the data or transaction log files.

In order to properly geo-replicate the data and maintain a consistent database image, the content of all nine VHDs would have to be geo-replicated in the exact order the I/O operations were executed against the nine different VHDs. However, Azure Storage geo-replication does not allow to declare dependencies between VHDs. This means Microsoft Azure Storage geo-replication doesn’t know about the fact that the content in these nine different VHDs are related to each other and that the data changes are consistent only when replicating in the order the I/O operations happened across all the 9 VHDs.

Besides chances being high that the geo-replicated images in the scenario do not provide a consistent database image, there also is a performance penalty that shows up with geo redundant storage that can severely impact performance. In summary do not use this type of storage redundancy for DBMS type workloads.

### Mapping VHDs into Azure Virtual Machine Service Storage Accounts

When placing VHDs in Azure Virtual Machine Services one needs to assign those to an Azure Storage Account. An Azure Storage Account is not only an administrative construct, but also is subject of limitations. Whereas the limitations vary on whether we talk about an Azure Standard Storage Account or an Azure Premium Storage Account. The exact capabilities and limitations are listed here: <http://azure.microsoft.com/en-us/documentation/articles/storage-scalability-targets/>

So for Azure Standard Storage it is important to note there is a limit on the IOPS per storage account (Column called ‘Total Request Rate’ in the article). In addition, there is a limit of 100 Storage Accounts per Azure subscription (as of July 2015). Therefore, it is recommended to balance IOPS of VMs between multiple storage accounts when using Azure Standard Storage. Whereas a single VM ideally uses one storage account if possible. So if we talk about DBMS deployments where each VHD that is hosted on Azure Standard Storage could reach its quota limit, you should only deploy 30-40 VHDs per Azure Storage Account that uses Azure Standard Storage. On the other hand, if you leverage Azure Premium Storage and want to store large database volumes, you might be fine in terms of IOPS. But an Azure Premium Storage Account is way more restrictive in data volume than an Azure Standard Storage Account. As a result, you can only deploy a limited number of VHDs within an Azure Premium Storage Account before hitting the data volume limit. At the end think of an Azure Storage Account as a “Virtual SAN” that has limited capabilities in IOPS and/or capacity. As a result, the task remains, as in on-premises deployments, to define the layout of the VHDs of the different SAP systems over the different ‘imaginary SAN devices’ or Azure Storage Accounts.

For Azure Standard Storage it is recommended not to present storage from different storage accounts to a single VM if possible.

Whereas using the DS-series of Azure VMs it is possible to mount VHDs out of Azure Standard Storage Accounts and Premium Storage Accounts. Use cases like writing backups into Standard Storage backed VHDs whereas having DBMS data and log files on Premium Storage come to mind where such heterogeneous storage could be leveraged.

Based on customer deployments and testing around 30 to 40 VHDs containing database data files and log files can be provisioned on a single Azure Standard Storage Account with acceptable performance. As mentioned earlier, the limitation of an Azure Premium Storage Account is likely to be the data capacity it can hold and not IOPS.

As with SAN devices on-premises, sharing requires some monitoring in order to eventually detect bottlenecks on an Azure Storage Account. The Azure Monitoring Extension for SAP and the Azure Portal are tools that can be used to detect busy Azure Storage Accounts that may be delivering suboptimal IO performance. If this situation is detected it is recommended to move busy VMs to another Azure Storage Account. Please refer to the ‘*SAP NetWeaver on Microsoft Azure Virtual Machine Services – Deployment Guide’* for details on how to activate the SAP host monitoring capabilities.

Another article summarizing best practices around Azure Standard Storage and Azure Standard Storage Accounts can be found here: <http://blogs.msdn.com/b/mast/archive/2014/10/14/configuring-azure-virtual-machines-for-optimal-storage-performance.aspx> .

### Moving deployed DBMS VMs from Azure Standard Storage to Azure Premium Storage

We encounter quite some scenarios where you as customer want to move a deployed VM from Azure Standard Storage into Azure Premium Storage. This is not possible without moving the data. There are several ways to achieve the goal:

* You could simply copy all VHDs, base VHD as well as data VHDs into a new Azure Premium Storage Account. Very often you chose the number of VHDs in Azure Standard Storage not because of the fact that you needed the data volume. But you needed that many VHDs because of the IOPS. Now that you move to Azure Premium Storage you could do with way less VHDs to achieve the some IOPS throughput. Given the fact that in Azure Standard Storage you pay for the used data and not the nominal disk size, the number of VHDs did not really matter in terms of costs. However, with Azure Premium Storage, you would pay for the nominal disk size. Therefore, most of the customer try to keep the number of Azure VHDs in Premium Storage at the number needed to achieve the IOPS throughput necessary. So, most customers decide against this way of a simple 1:1 copy.
* If not yet mounted, you mount a single VHD that can contain a database backup of your SAP database. After the backup, you unmount all VHDs including the VHD containing the backup and copy the base VHD the VHD with the backup into an Azure Premium Storage account. There you deploy the VM based on the base VHD. You then mount the VHD with the backup. Now you create additional empty Premium Storage Disks for the VM that are used to restore the database into. This assumes that the DBMS allows you to change paths to the data and log files as part of the restore process.
* Another possibility is a variation of the former process, where you just copy the backup VHD into Azure Premium Storage and attach it against a VM that you newly deployed and installed.
* The fourth possibility you would choose when you are in need to change the number of datafiles of your database. In such a case you would perform a SAP homogenous system copy using export/import. Put those export files into a VHD that is copied into an Azure Premium Storage Account. There it is getting attached to a VM that you use to run the import processes. Customers use this possibility mainly when they want to decrease the number of data files.

## Deployment of VMs for SAP in Azure

Microsoft Azure offers multiple ways to deploy VMs and associated disks. Thereby it is very important to understand the differences since preparations of the VMs might differ dependent on the way of deployment. In general, we look into the following scenarios:

* Deploying a VM out of the Azure Gallery  
  You like to take a Microsoft or 3rd party provided image out of the Azure Gallery to deploy your VM. After you deployed your VM in Azure, you follow the same guidelines and tools to install the SAP software inside your VM as you would do in an on-premises environment. For installing the SAP software inside the Azure VM, SAP and Microsoft recommend to upload and store the SAP installation media in Azure VHDs or to create an Azure VM working as a ‘File server’ which contains all the necessary SAP installation media.
* Deploying a VM with a customer specific generalized image  
  Due to specific patch requirements in regards to your OS or DBMS version, the provided images out of the Azure Gallery might not fit your needs. Therefore, you might need to create a VM using your own ‘private’ OS/SQL Server VM image which can be deployed several times afterwards.   
  To prepare such a ‘private’ image for duplication, the Windows settings (like Windows SID and hostname) must be abstracted/generalized on the on-premises VM; this can be done using [sysprep](http://technet.microsoft.com/en-us/library/cc721940(v=ws.10).aspx). If you have already installed SAP content in your on-premises VM (especially for 2-Tier systems), you can adapt the SAP system settings after the deployment of the Azure VM through the instance rename procedure supported by the SAP Software Provisioning Manager (SAP Note [1619720 - System Rename for SAP Systems based on SAP NetWeaver](http://service.sap.com/sap/support/notes/1619720) – login required).  
  Otherwise you can install the SAP software later after the deployment of the Azure VM.   
  As of the database content used by the SAP application, you can either generate the content freshly by an SAP installation or you can import your content into Azure by using a VHD with a DBMS database backup or by leveraging capabilities of the DBMS to directly backup into Microsoft Azure Storage. In this case, you could also prepare VHDs with the DBMS data and log files on-premises and then import those as Disks into Azure. But the transfer of DBMS data which is getting loaded from on-premises to Azure would work over VHD disks that need to be prepared on-premises.
* Moving a VM from on-premises to Azure with a non-generalized disk  
  You plan to move a specific SAP system from on-premises to Azure (lift and shift). This can be done by uploading the VHD which contains the OS, the SAP Binaries and eventual DBMS binaries plus the VHDs with the data and log files of the DBMS to Azure. In opposite to scenario #2 above, you keep the hostname, SAP SID and SAP user accounts in the Azure VM as they were configured in the on-premises environment. Therefore, preparing the image by calling sysprep is not necessary. This case will mostly apply for Cross-Premises scenarios where a part of the SAP landscape is run on-premises and parts on Azure.

# High Availability and Disaster Recovery with Azure VMs

Azure offers the following High Availability (HA) and Disaster Recovery (DR) functionalities which apply to different components we would use for SAP and DBMS deployments

## VMs deployed on Azure Nodes

The Azure Platform does not offer features such as Live Migration for deployed VMs. This means if there is maintenance necessary on a server cluster on which a VM is deployed, the VM needs to get stopped and restarted. Maintenance in Azure is performed using so called Upgrade Domains within clusters of servers. Only one Upgrade Domain at a time is being maintained. During such a restart there will be an interruption of service while the VM is shut down, maintenance is performed and VM restarted. Most DBMS vendors however provide High Availability and Disaster Recovery functionality that will quickly restart the DBMS services on another node if the primary node is unavailable. The Azure Platform offers functionality to distribute VMs, Storage and other Azure services across Upgrade Domains to ensure that planned maintenance or infrastructure failures would only impact a small subset of VMs or services. With careful planning it is possible to achieve availability levels comparable to on-premises infrastructures.

Microsoft Azure Availability Sets are a logical grouping of VMs or Services that ensures VMs and other services are distributed to different Fault and Upgrade Domains within a cluster such that there would only be one node shutdown at any one point in time (see details: <http://www.windowsazure.com/en-us/documentation/articles/manage-availability-virtual-machines/>). It needs to be configured by purpose when rolling out VMs as seen here:

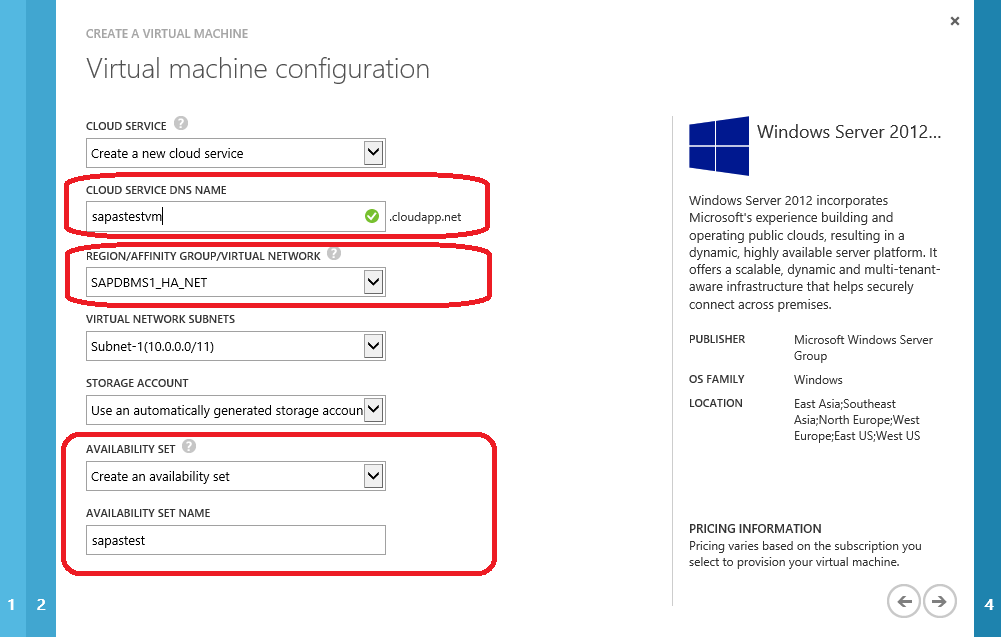


Figure Definition of Availability Set for DBMS HA configurations

If we want to create highly available configurations of DBMS deployments (independent of the individual DBMS HA functionality used), the DBMS VMs would need to:

* Belong to the same Cloud Service (<http://www.windowsazure.com/en-us/documentation/services/cloud-services/>). One should not add a load-balancing set to the VMs, but just keep the VMs with their different private ports in the same Cloud Service. The reason for having the VMs in the same service is to enable Domain Name Resolution. In Cloud-Only deployments and if the VMs are not joined to a domain, name resolution would not work if the two VMs would be in different Azure Cloud Services, but still in the same Azure Virtual Network (<http://www.windowsazure.com/en-us/documentation/services/virtual-network/>). Only IP resolution would work under such circumstances which would not be sufficient. Hence the same Cloud Service is required. In Cross-Premises scenarios, the fact whether the VMs are in the same or a different Cloud Service has no impact on name and IP resolution since that is done via the on-premises AD service and not the Azure AD service
* Add the VMs to the same Azure Virtual Network
* The VMs of the HA configuration should also be in the same subnet. Name resolution between the different subnets is not possible in Cloud-Only deployments, only IP resolution will work. Using site-to-site or ExpressRoute connectivity for Cross-Premises deployments, a network with at least one subnet will be already established. Name resolution will be done according to the on-premises AD policies and network infrastructure.

### IP Addresses

It is highly recommended to setup the VMs for HA configurations in a resilient way. Relying on IP addresses to address the HA partner(s) within the HA configuration is not reliable in Azure unless static IP addresses are used. There are two “Shutdown” concepts in Azure:

* Shut down through Azure Portal or Azure PowerShell cmdlet **Stop-AzureVM**: In this case the Virtual Machine gets shutdown and de-allocated. Your Azure account will no longer be charged for this VM so the only charges that will incur is for the storage used. However, if the last VM of a Cloud Service is shut down AND de-allocated, the IP addresses within the Virtual Network or the subnet are getting released and are not guaranteed to be assigned at the restart of the VM. Performing the shut down through the Azure Portal will also automatically cause de-allocation. But when using the **Stop-AzureVM** cmdlet, one must use the *–FORCE* option to force de-allocation.
* Shut down within the VM or Azure PowerShell cmdlet **Stop-AzureVM**: In this case the VM gets shut down and NOT de-allocated. However, in this case, your Azure account will still be charged for the VM, despite the fact that it is shutdown. In such a case, the assignment of the IP address to a stopped VM will remain intact. Shutting down the VM from within will not automatically force de-allocation. And when using the *Stop-AzureVM* cmdlet, one would use the cmdlet w/o *the –FORCE* option to ensure it is not de-allocated.

Even for Cross-Premises scenarios, by default a shutdown and de-allocation will mean de-assignment of the IP addresses from the VM, even if on-premises policies in DHCP settings are different.

* The exception is if one assigns fixed IP addresses to VMs as described here: <http://msdn.microsoft.com/en-us/library/windowsazure/dn630228.aspx>  
  Please note the usage of fixed IP addresses requires the usage of Azure Virtual Networks.
* In such a case the IP address remains fixed as long as the VM is not deleted then the IP address gets de-assigned.

**Recommendation:** In order to keep the whole deployment simple and manageable, the clear recommendation is to setup the VMs partnering in a DBMS HA or DR configuration within Azure in a way that there is a functioning name resolution between the different VMs involved.

# Deployment of SAP related and required Host Monitoring capabilities

For productive usage of SAP Applications in Azure Virtual Machine Services, SAP requires the ability to get host monitoring data from the physical hosts running the Azure Virtual Machines. A specific SAP HostAgent patch level will be required that enables this capability in SAPOSCOL and SAP HostAgent. The exact patch level is documented in SAP Note [1409604 – Virtualization on Windows: Enhanced Monitoring](http://service.sap.com/sap/support/notes/1409604) .

For the details regarding deployment of components that deliver host data to SAPOSCOL and SAPHostAgent and the lifecycle management of those components, please refer to the document: ‘*SAP NetWeaver on Microsoft Azure Virtual Machine Services – Deployment Guide*’

# Specifics to SQL Server RDBMS

## SQL Server IaaS

Starting with Microsoft Azure, you can easily migrate your existing SQL Server applications built on Windows Server platform to Azure Virtual Machines. SQL Server in a Virtual Machine enables you to reduce the total cost of ownership of deployment, management and maintenance of enterprise breadth applications by easily migrating these applications to Microsoft Azure. With SQL Server in an Azure Virtual Machine, administrators and developers can still use the same development and administration tools that are available on-premises.

**IMPORTANT: Please note we are not discussing Microsoft Azure SQL Database which is a Platform as a Service offer of the Microsoft Azure Platform. The discussion in this paper is about running the SQL Server product as it is known for on-premises deployments in Azure Virtual Machines, leveraging the Infrastructure as a Service capability of Azure. Database capabilities and functionalities between these two offers are different and should not be mixed up with each other. See also:** [**http://www.windowsazure.com/en-us/services/data-management/**](http://www.windowsazure.com/en-us/services/data-management/)

It is strongly recommended to review this documentation before continuing: <http://msdn.microsoft.com/library/azure/jj823132.aspx> .

In the following sections pieces of parts of the documentation under the link above will be aggregated and mentioned. Specifics around SAP are mentioned as well and some concepts are described in more detail. However, it is highly recommended to work through the documentation above first before reading the SQL Server specific documentation.

There is some SQL Server in IaaS specific information you should know before continuing:

* **Virtual Machine SLA:** There is an SLA for the Azure Virtual Machines which can be found here: <http://www.windowsazure.com/en-us/support/legal/sla/>
* **SQL Version Support:** For SAP customers, we will support SQL Server 2014, 2012 and 2008 R2 on Microsoft Azure Virtual Machine. Earlier editions are not supported. Review this general [Support Statement [http://social.technet.microsoft.com/wiki/cfs-file.ashx/__key/communityserver-components-sitefiles/10_5F00_external.png](http://support.microsoft.com/kb/956893)](http://support.microsoft.com/kb/956893)for more details. Please note that in general SQL Server 2008 is supported by Microsoft as well. However due to significant functionality for SAP which was introduced with SQL Server 2008 R2, SQL Server 2008 R2 is the minimum release for SAP. Keep in mind that SQL Server 2012 and 2014 got extended with deeper integration into the IaaS scenario (like backing up directly against Azure Storage). Therefore, we restrict this paper to SQL Server 2012 and 2014 with its latest patch level for Azure.
* **SQL Feature Support:** Most SQL Server features are supported on Microsoft Azure Virtual Machine Services with some exceptions. **SQL Server Failover Clustering using Shared Disks is not supported**.  Distributed technologies like Database Mirroring, AlwaysOn Availability Groups, Replication, Log Shipping and Service Broker are supported within a single Azure Region. SQL Server AlwaysOn also is supported between different Azure Regions as documented here: <http://blogs.technet.com/b/dataplatforminsider/archive/2014/06/19/sql-server-alwayson-availability-groups-supported-between-microsoft-azure-regions.aspx>. Review the [Support Statement [http://social.technet.microsoft.com/wiki/cfs-file.ashx/__key/communityserver-components-sitefiles/10_5F00_external.png](http://support.microsoft.com/kb/956893)](http://support.microsoft.com/kb/956893)for more details.  Also, check out the Best Practices documented here: <https://msdn.microsoft.com/library/azure/jj823132.aspx>
* **SQL Performance:** We are confident that Microsoft Azure hosted Virtual Machines will perform very well in comparison to other public cloud virtualization offerings, but individual results may vary. Check out:
  + [Performance Best Practices for SQL Server in Azure Virtual Machines](https://msdn.microsoft.com/en-us/library/azure/dn133149.aspx)
* **Using Gallery Images:** The fastest way to deploy a new Microsoft Azure VMs is to use an image from the gallery.  There are images in the Azure gallery which contain SQL Server. The images where SQL Server already is installed can’t be immediately used for SAP NetWeaver applications. The reason is the default SQL Server collation is installed within those images and not the collation required by SAP NetWeaver systems. In order to use such images, please check the steps documented in [chapter 5.](#_Using_a_SQL)6 of this document.
* Check out [Pricing Details [http://social.technet.microsoft.com/wiki/cfs-file.ashx/__key/communityserver-components-sitefiles/10_5F00_external.png](https://www.windowsazure.com/en-us/pricing/details/)](https://www.windowsazure.com/en-us/pricing/details/)for more information. The [SQL Server 2012 Licensing Guide [http://social.technet.microsoft.com/wiki/cfs-file.ashx/__key/communityserver-components-sitefiles/10_5F00_external.png](http://download.microsoft.com/download/4/F/7/4F74E127-827E-420D-971F-53CECB6778BD/SQL_Server_2012_Licensing_Datasheet_and_FAQ_Mar2012.docx)](http://download.microsoft.com/download/7/3/C/73CAD4E0-D0B5-4BE5-AB49-D5B886A5AE00/SQL_Server_2012_Licensing_Reference_Guide.pdf)and [SQL Server 2014 Licensing Guide](http://download.microsoft.com/download/B/4/E/B4E604D9-9D38-4BBA-A927-56E4C872E41C/SQL_Server_2014_Licensing_Guide.pdf) are also an important resource.

## SQL Server configuration guidelines for SAP related SQL Server installations in Azure VMs

### Recommendations on VM/VHD structure for SAP related SQL Server deployments

In accordance with the general description, SQL Server executables should be located or installed into the system drive of the VM’s base VHD (drive C:\). Typically, most of the SQL Server system databases are not utilized at a high level by SAP NetWeaver workload. Hence the system databases of SQL Server (master, msdb, and model) can remain on the C:\ drive as well. An exception could be tempdb, which in the case of some SAP ERP and all BW workloads, might require either higher data volume or I/O operations volume which can’t fit into the original VM. For such systems, the following steps should be performed:

* Move the primary tempdb data file(s) to the same logical drive as the primary data file(s) of the SAP database.
* Add any additional tempdb data files to each of the other logical drives containing a data file of the SAP user database.
* Add the tempdb logfile to the logical drive which contains the user database’s log file.
* **Exclusively for VM types that use local SSDs** on the compute node tempdb data and log files might be placed on the D:\ drive. Nevertheless, it might be recommended to use multiple tempdb data files. Be aware D:\ drive volumes are different based on the VM type.

These configurations enable tempdb to consume more space than the system drive is able to provide. In order to determine the proper tempdb size, one can check the tempdb sizes on existing systems which run on-premises. In addition, such a configuration would enable IOPS numbers against tempdb which cannot be provided with the system drive. Again, systems that are running on-premises can be used to monitor I/O workload against tempdb so that you can derive the IOPS numbers you expect to see on your tempdb.

A VM configuration which runs SQL Server with a SAP database and where tempdb data and tempdb logfile are placed on the D:\ drive would look like:



Figure Reference Configuration of Azure IaaS VM for SAP

Be aware that the D:\ drive has different sizes dependent on the VM type. Dependent on the size requirement of tempdb you might be forced to pair tempdb data and log files with the SAP database data and log files in cases where D:\ drive is too small.

### Formatting the VHDs

For SQL Server the NTFS block size for VHDs containing SQL Server data and log files should be 64K. There is no need to format the D:\ drive. This drive comes pre-formatted.

In order to make sure that the restore or creation of databases is not initializing the data files by zeroing the content of the files, one should make sure that the user context the SQL Server service is running in has a certain permission. Usually users in the Windows Administrator group have these permissions. If the SQL Server service is run in the user context of non-Windows Administrator user, you need to assign that user the User Right ‘Perform volume maintenance tasks’. See the details in this Microsoft Knowledge Base Article: <http://support.microsoft.com/kb/2574695>

### Impact of database compression

In configurations where I/O bandwidth can become a limiting factor, every measure which reduces IOPS might help to stretch the workload one can run in an IaaS scenario like Azure. Therefore, if not yet done, applying SQL Server PAGE compression is strongly recommended by both SAP and Microsoft before uploading an existing SAP databases to Azure.

The recommendation to perform Database Compression before uploading to Azure is given out of two reasons:

* The amount of data to be uploaded is lower.
* The duration of the compression execution is shorter assuming that one can use stronger hardware with more CPUs or higher I/O bandwidth or less I/O latency on-premises.

Database compression works as well in a VM hosted in Azure Virtual Machine Services as it does on-premises. For more details on how to compress an existing SAP SQL Server database please check here: <http://blogs.msdn.com/b/saponsqlserver/archive/2010/10/08/compressing-an-sap-database-using-report-msscompress.aspx>

## SQL Server 2014 – Storing Database Files directly on Azure Blog Storage

SQL Server 2014 opens the possibility to store database files directly on Azure Blob Store without the ‘wrapper’ of a VHD around them. Especially with using Standard Azure Storage or smaller VM types this enables scenarios where you can overcome the limits of IOPS that would be enforced by a limited number of VHDs that can be mounted to some smaller VM types. This works for user databases however not for system databases of SQL Server. It also works for data and log files of SQL Server. If you’d like to deploy a SAP SQL Server database this way instead of ‘wrapping’ it into VHDs, please keep the following in mind:

* The Storage Account used needs to be in the same Azure Region as the one that is used to deploy the VM SQL Server is running in.
* Considerations listed earlier in regards to distribute VHDs over different Azure Storage Accounts are applying for this method of deployments as well. Means the I/O operations count against the limits of the Azure Storage Account.
* So far this way of deployment does not work against Azure Premium Storage.

Details about this type of deployment are listed here: <https://msdn.microsoft.com/en-us/library/dn385720.aspx>

In order to store SQL Server data files directly on Azure Premium Storage, you need to have a minimum SQL Server 2014 patch release which is documented here: <https://support.microsoft.com/en-us/kb/3063054> . Storing SQL Server data files on Azure Standard Storage does work with the released version of SQL Server 2014. However, the very same patches contain another series of fixes which make the direct usage of Azure Blob Storage for SQL Server data files and backups more reliable. Therefore we recommend to use these patches in general.

## SQL Server 2014 Buffer Pool Extension

SQL Server 2014 introduced a new feature which is called Buffer Pool Extension. This functionality extends the buffer pool of SQL Server which is kept in memory with a second level cache that is backed by local SSDs of a server or VM. This enables to keep a larger working set of data ‘in memory’. Compared to accessing Azure Standard Storage the access into the extension of the buffer pool which is stored on local SSDs of an Azure VM is many factors faster. Therefore, leveraging the local D:\ drive of the VM types that have excellent IOPS and throughput (A0 to A7) could be a very reasonable way to reduce the IOPS load against Azure Storage and improve response times of queries dramatically. This applies especially when not using Premium Storage. In case of Premium Storage and the usage of the Premium Azure Read Cache on the compute node, as recommended for data files, no big differences are expected. Reason is that both caches (SQL Server Buffer Pool Extension and Premium Storage Read Cache) are using the local disks of the compute nodes.

For more details about this functionality, please check this documentation: <https://msdn.microsoft.com/en-us/library/dn133176.aspx>

## Backup/Recovery considerations for SQL Server

When deploying SQL Server into Azure your backup methodology must be reviewed. Even if the system is not a productive system, the SAP database hosted by SQL Server must be backed up periodically. Since Azure Storage keeps three images, a backup is now less important in respect to compensating a storage crash. The priority reason for maintaining a proper backup and recovery plan is more that you can compensate for logical/manual errors by providing point in time recovery capabilities. So the goal is to either use backups to restore the database back to a certain point in time or to use the backups in Azure to seed another system by copying the existing database. For example, you could transfer from a 2-Tier SAP configuration to a 3-Tier system setup of the same system by restoring a backup.

There are three different ways to backup SQL Server to Azure Storage:

1. SQL Server 2012 CU4 and higher can natively backup databases to a URL. This is detailed in the blog: [**New functionality in SQL Server 2014 – Part 5 – Backup/Restore Enhancements**](http://blogs.msdn.com/b/saponsqlserver/archive/2014/02/15/new-functionality-in-sql-server-2014-part-5-backup-restore-enhancements.aspx). See chapter 5.5.1
2. SQL Server releases prior to SQL 2012 CU4 can use a redirection functionality to backup to a VHD and basically move the write stream towards an Azure Storage location that has been configured. See chapter 5.5.2
3. The final method is to perform a conventional SQL Server backup to disk command onto a VHD disk device. This is identical to the on-premises deployment pattern and is not discussed in detail in this document.

### SQL Server 2012 SP1 CU4 and later

This functionality allows you to directly backup to Azure BLOB storage. Without this method, you must backup to other Azure VHDs which would consume VHD and IOPS capacity. The idea is basically this:



Figure Using SQL Server 2012 Backup to Microsoft Azure Storage BLOB

The advantage in this case is that one doesn’t need to spend VHDs to store SQL Server backups on. So you have fewer VHDs allocated and the whole bandwidth of VHD IOPS can be used for data and log files. Please note that the maximum size of a backup is limited to a maximum of 1 TB as documented in the section ‘Limitations’ in this article: <http://msdn.microsoft.com/en-us/library/jj919148.aspx>. If the backup size, despite using SQL Server Backup compression would exceed 1 TB in size, the functionality described in the [chapter 5.5.2](#_SQL_Server_2012) in this document needs to be used.

Related documentation describing the restore of databases from backups against Azure Blob Store: <http://msdn.microsoft.com/en-us/library/dn449492.aspx> recommend not to restore directly from Azure BLOB store if the backups is >25GB. The recommendation in this article is simply based on performance considerations and not due to functional restrictions. Therefore, different conditions may apply on a case by case basis.

Documentation on how this type of backup is set up and leveraged can be found in this tutorial:

<http://msdn.microsoft.com/en-us/library/jj720558.aspx>

An example of the sequence of steps can be read here:

<http://msdn.microsoft.com/en-us/library/jj919148.aspx#complete>

Automating backups, it is of highest importance to make sure that the BLOBs for each backup are named differently. Otherwise they will be overwritten and the restore chain is broken.

In order not to mix up things between the 3 different types of backups, it is advisable to create different containers underneath the storage account used for backups. The containers could be by VM only or by VM and Backup type. The schema could look like:

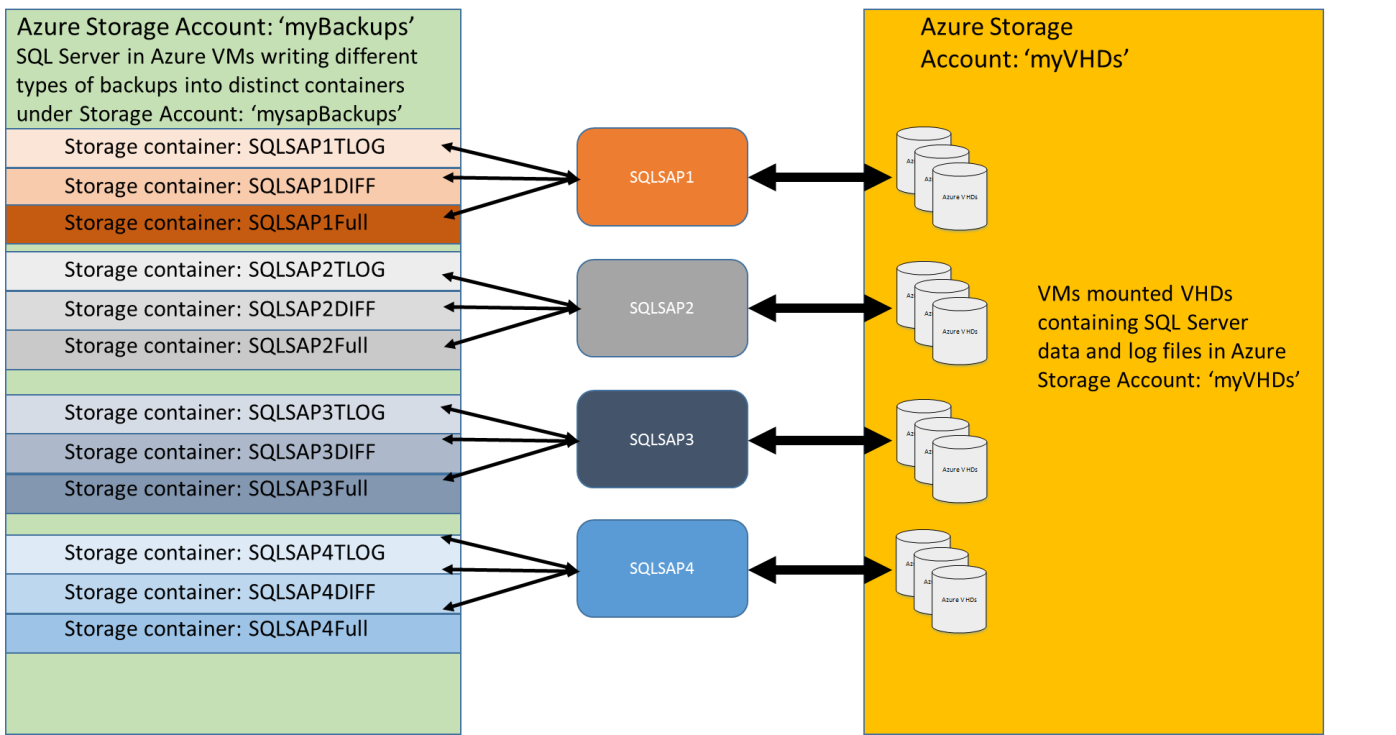


Figure Using SQL Server 2012 Backup to Microsoft Azure Storage BLOB – Different containers under separate Storage Account

In the example above, the backups would not be performed into the same storage account where the VMs are deployed. There would be a new storage account specifically for the backups. Within the storage accounts, there would be different containers created with a matrix of the type of backup and the VM name. Such segmentation will make it easier to administrate the backups of the different VMs.

The BLOBs one directly writes the backups to, are not adding to the count of the VHDs of a VM. Hence one could maximize the maximum of VHDs mounted of the specific VM SKU for the data and transaction log file and still execute a backup against a storage container.

### SQL Server 2012 SP1 CU3 and earlier releases

The first step you must perform in order to achieve a backup directly against Azure Storage would be to download the msi which is linked to this KBA article: <http://www.microsoft.com/en-us/download/details.aspx?id=40740>

Download the x64 installation file and the documentation. The file will install a program called: ‘Microsoft SQL Server Backup to Microsoft Azure Tool’. Read the documentation of the product thoroughly. The tool basically works in the following way:

* From the SQL Server side, a disk location for the SQL Server backup is defined (don’t use the D:\ drive for this).
* The tool will allow you to define rules which can be used to direct different types of backups to different Azure Storage containers.
* Once the rules are in place, the tool will redirect the write stream of the backup to one of the VHDs/disks to the Azure Storage location which was defined earlier.
* The tool will leave a small stub file of a few KB size on the VHD/Disk which was defined for the SQL Server backup. **This file should be left on the storage location since it is required to restore again from Azure Storage.**
  + If you have lost the stub file (e.g. through loss of the storage media that contained the stub file) and you have chosen the option of backing up to a Windows Azure Storage account, you may recover the stub file through Windows Azure Storage by downloading it from the storage container in which it was placed. You should then place the stub file into a folder on the local machine where the Tool is configured to detect and upload to the same container with the same encryption password if encryption was used with the original rule.

This means the schema as described above for more recent releases of SQL Server can be put in place as well for SQL Server releases which are not allowing direct address an Azure Storage location.

This method should not be used with more recent SQL Server releases which support backing up natively against Azure Storage. Exceptions are where limitations of the native backup into Azure are blocking native backup execution into Azure.

### Other possibilities to backup SQL Server databases

Other possibilities to backup databases is to attach additional VHDs to a VM that you use to store backups on. In such a case you would need to make sure that the VHDs are not running full. If that is the case, you would need to unmount the VHD and so to speak ‘archive’ it and replace it with a new empty VHD. If you go down that path, you want to keep these VHDs in separate Azure Storage Accounts from the ones that the VHDs with the database files.

A second possibility is to use a large VM that can have many VHDs attached. E.g. D14 with 32VHDs. Use Storage Spaces to build a flexible environment where you could build shares that are used then as backup targets for the different DBMS servers.

Some practices got documented here as well: <http://blogs.msdn.com/b/sqlcat/archive/2015/02/26/large-sql-server-database-backup-on-an-azure-vm-and-archiving.aspx>

### Performance considerations for backups/restores

As in bare-metal deployments, backup/restore performance is dependent on how many volumes can be read in parallel and what the throughput of those volumes might be. In addition, the CPU consumption used by backup compression may play a significant role on VMs with just up to 8 CPU threads. Therefore, one can assume:

* The fewer the number of VHDs used to store the data files, the smaller the overall throughput in reading.
* The smaller the number of CPU threads in the VM, the more severe the impact of backup compression.
* The fewer targets (BLOBs or VHDs) to write the backup to, the lesser the throughput.
* The smaller the VM size, the smaller the storage throughput quota writing and reading from Azure Storage. Independent of whether the backups are directly stored on Azure Blob or whether they are stored in VHDs that again are stored in Azure Blobs.

When using a Microsoft Azure Storage BLOB as the backup target in more recent releases, you are restricted to designating only one URL target for each specific backup.

But when using the ‘Microsoft SQL Server Backup to Microsoft Azure Tool’ in older releases, you can define more than one file target. With more than one target, the backup can scale and the throughput of the backup is higher. This would result then in multiple files as well in the Azure Storage account. In our testing, using multiple file destinations one can definitely achieve the throughput which one could achieve with the backup extensions implemented in from SQL Server 2012 SP1 CU4 on. You also are not blocked by the 1TB limit as in the native backup into Azure.

However, keep in mind, the throughput also is dependent on the location of the Azure Storage Account you use for the backup. An idea might be to locate the storage account in a different region than the VMs are running in. E.g. you would run the VM configuration in West-Europe, but put the Storage Account that you use to back up against in North-Europe. That certainly will have impact on the backup throughput and is not likely to generate a throughput of 150MB/sec as it seems to be possible in cases where the target storage and the VMs are running in the same regional datacenter.

### Managing Backup BLOBs

There is a requirement to manage the backups on your own. Since the expectation is that many blobs will be created by executing frequent transaction log backups, administration of those blobs easily can overburden the Azure Portal. Therefore, it is recommendable to leverage a 3rd party Azure Storage Explorer. There are several good ones available which can help to manage an Azure storage account. There is a codeplex project which can be found here:

<http://azurestorageexplorer.codeplex.com/downloads/get/391105>

There also are professional editions of Azure Storage Explorers which can be found here:

<http://www.cerebrata.com/>

<http://clumsyleaf.com/products/cloudxplorer>

### Azure VM backup

VMs within the SAP system can be backed up using Azure Virtual Machine Backup functionality. Azure Virtual Machine Backup got introduced early in the year 2015 and meanwhile is a standard method to backup a complete VM in Azure. Azure Backup stores the backups in Azure and allows a restore of a VM again.

VMs that run databases can be backed up in a consistent manner as well if the DBMS systems supports the Windows VSS (Volume Shadow Copy Service - <https://msdn.microsoft.com/en-us/library/windows/desktop/bb968832(v=vs.85).aspx> ) as SQL Server does. So using Azure VM backup could be a way to get to a restorable backup of a SAP database. However, be aware that based on Azure VM backups point-in-time restores of databases is not possible. Therefore, the recommendation is to perform backups of databases with DBMS functionality instead of relying on Azure VM Backup.

To get familiar with Azure Virtual Machine Backup please start here: <http://azure.microsoft.com/en-us/documentation/services/backup/> .

## Using a SQL Server images out of the Microsoft Azure Gallery

Microsoft offers VMs in the general gallery which already contains versions of SQL Server. For SAP customers who require licenses for SQL Server and Windows, this might be an opportunity to basically cover the need for licenses by spinning up VMs with SQL Server already installed. In order to use such images for SAP, the following considerations need to be made:

* The SQL Server non-Evaluation versions acquire higher costs than just a ‘Windows-only’ VM out of the gallery. Please see these articles to compare prices: <http://www.windowsazure.com/en-us/pricing/details/virtual-machines/> and <http://www.windowsazure.com/en-us/pricing/details/virtual-machines/#service-sql-server>.
* You only can use SQL Server releases which are supported by SAP, like SQL Server 2012.
* The collation of the SQL Server instance which is installed in the VMs offered in the Gallery is not the collation SAP NetWeaver requires the SQL Server instance to run. You can change the collation though with the directions in the following section.

### Changing the SQL Server Collation of a Microsoft Windows/SQL Server VM

Since the Gallery image of a Windows/SQL Server Gallery VM is not what SAP NetWeaver applications require, it needs to be changed immediately after the deployment. For SQL Server 2012, this can be done with the following steps as soon as the VM has been deployed and an administrator is able to log into the deployed VM:

* Open a Windows Command Window ‘as administrator’.
* Change the directory to C:\Program Files\Microsoft SQL Server\110\Setup Bootstrap\SQLServer2012.
* Execute the command: Setup.exe /QUIET /ACTION=REBUILDDATABASE /INSTANCENAME=MSSQLSERVER /SQLSYSADMINACCOUNTS=<local\_admin\_account\_name> /SQLCOLLATION=SQL\_Latin1\_General\_Cp850\_BIN2
  + <local\_admin\_account\_name> is the account which was defined as the administrator account when deploying the VM for the first time through the gallery.

The process should take between 2 and 5minutes. In order to make sure whether the step ended up with the correct result, please perform the following steps:

* Open SQL Server Management Studio.
* Open a Query Window.
* Execute the command sp\_helpsort in the SQL Server master database.

The desired result should look like:

Latin1-General, binary code point comparison sort for Unicode Data, SQL Server Sort Order 40 on Code Page 850 for non-Unicode Data

If this is not the result, **STOP** deploying SAP and investigate why the setup command did not work as expected. Deployment of SAP NetWeaver applications onto SQL Server instance with different SQL Server codepages than the one mentioned above is not supported.

## SQL Server High-Availability for SAP in Azure

As mentioned earlier in this paper, there is no possibility to create shared storage which is necessary for the usage of the oldest SQL Server high availability functionality. This functionality would install two or more SQL Server instances in a Windows Server Failover Cluster (WSFC) using a shared disk for the user databases (and eventually tempdb). This is the long time standard high availability method which also is supported by SAP. Because Azure doesn’t support shared storage, SQL Server high availability configurations with a shared disk cluster configuration cannot be realized. However, many other high availability methods are still possible and are described in the following sections.

Before reading the different specific high availability technologies usable for SQL Server in Azure, there is a very good document which gives more details and pointers here: <http://msdn.microsoft.com/en-us/library/windowsazure/jj870962.aspx>

### SQL Server Log Shipping

One of the methods of high availability (HA) is SQL Server Log Shipping. If the VMs participating in the HA configuration have working name resolution, there is no problem and the setup in Azure will not differ from any setup that is done on-premises. It is not recommended to rely on IP resolution only. In regards to setting up Log Shipping and the principles around Log Shipping please check this documentation:

<http://technet.microsoft.com/en-us/library/ms187103.aspx>

In order to really achieve any high availability, one needs to deploy the VMs which are within such a Log Shipping configuration to be within the same Azure Availability Set.

### Database Mirroring

Database Mirroring as supported by SAP (see SAP Note [965908 – SQL Server Database mirroring and SAP Applications](http://service.sap.com/sap/support/notes/965908) ) relies on defining a failover partner in the SAP connection string. For the Cross-Premises cases, we assume that the two VMs are in the same domain and that the user context the two SQL Server instances are running under are domain users as well and have sufficient privileges in the two SQL Server instances involved. Therefore, the setup of Database Mirroring in Azure does not differ between a typical on-premises setup/configuration.

As of Cloud-Only deployments, the easiest method is to have another domain setup in Azure to have those DBMS VMs (and ideally dedicated SAP VMs) within one domain.

If a domain is not possible, one can also use certificates for the database mirroring endpoints as described here: <http://technet.microsoft.com/en-us/library/ms191477.aspx>

A tutorial to set-up Database Mirroring in Azure can be found here: <http://msdn.microsoft.com/en-us/library/windowsazure/jj870961.aspx>

### AlwaysOn

As AlwaysOn is supported for SAP on-premises (see SAP Note [1772688 – SQL Server AlwaysOn and SAP Applications](http://service.sap.com/sap/support/notes/1772688) ), it is supported to be used in combination with SAP in Azure. The fact that you are not able to create shared disks in Azure doesn’t mean that one can’t create an AlwaysOn Windows Server Failover Cluster (WSFC) configuration between different VMs. It only means that you do not have the possibility to use a shared disk as a quorum in the cluster configuration. Hence you can build an AlwaysOn WSFC configuration in Azure and simply not select the quorum type that utilizes shared disk. The Azure environment those VMs are deployed in should resolve the VMs by name and the VMs should be in the same domain. This is true for Azure only and Cross-Premises deployments. There are some special considerations around deploying the SQL Server Availability Group Listener (not to be confused with the Azure Availability Set) since Azure at this point in time does not allow to simply create an AD/DNS object as it is possible on-premises. Therefore, some different installation steps are necessary to overcome the specific behavior of Azure.

Some considerations using an Availability Group Listener are:

* Using an Availability Group Listener is only possible with Windows Server 2012 or Windows Server 2012 R2 as guest OS of the VM. For Windows Server 2012 you need to make sure that this patch is applied: <http://support.microsoft.com/kb/2854082>
* For Windows Server 2008 R2 this patch does not exist and AlwaysOn would need to be used in the same manner as Database Mirroring by specifying a failover partner in the connections string (done through the SAP default.pfl parameter dbs/mss/server – see SAP Note 965908).
* When using an Availability Group Listener, the Database VMs need to be isolated from the SAP application layer in a different Azure Cloud Service. Unless you have an AD and DNS deployed in Azure, in Cloud-Only deployments, this would require from an SAP application layer the maintenance of the etc\host file in order to get the VM names of the SQL Server VMs resolved. In order to avoid that Azure is assigning new IP addresses in cases where both VMs incidentally are shutdown, one should assign static IP addresses to those VMs in the AlwaysOn configuration (<http://msdn.microsoft.com/en-us/library/windowsazure/dn630228.aspx> )
* There are special steps required when building the WSFC cluster configuration where the cluster needs a special IP address assigned, because Azure with its current functionality would assign the cluster name the same IP address as the node the cluster is created on. This means a manual step must be performed to assign a different IP address to the cluster.
* The Availability Group Listener is going to be created in Azure with TCP/IP endpoints which are assigned to the VMs running the primary and secondary replicas of the Availability group.
* There might be a need to secure these endpoints with ACLs.

The detailed steps and necessities of installing an AlwaysOn configuration on Azure are best experienced when walking through the tutorial available here:

* <http://msdn.microsoft.com/en-us/library/dn249504.aspx>

Creating an Availability Group Listener is best described in this tutorial:

* <http://msdn.microsoft.com/en-us/library/dn376546.aspx>

Securing network endpoints with ACLs are explained best here:

* <http://michaelwasham.com/windows-azure-powershell-reference-guide/network-access-control-list-capability-in-windows-azure-powershell/>
* <http://blogs.technet.com/b/heyscriptingguy/archive/2013/08/31/weekend-scripter-creating-acls-for-windows-azure-endpoints-part-1-of-2.aspx>
* <http://blogs.technet.com/b/heyscriptingguy/archive/2013/09/01/weekend-scripter-creating-acls-for-windows-azure-endpoints-part-2-of-2.aspx>
* <http://blogs.technet.com/b/heyscriptingguy/archive/2013/09/18/creating-acls-for-windows-azure-endpoints.aspx>

It is possible to deploy a SQL Server AlwaysOn Availability Group over different Azure Regions as well. This functionality will leverage the Azure VNet-to-Vnet connectivity (<https://azure.microsoft.com/en-us/documentation/articles/virtual-networks-configure-vnet-to-vnet-connection/>). The setup of SQL Server AlwaysOn Availability Groups in such a scenario is described here: <http://blogs.technet.com/b/dataplatforminsider/archive/2014/06/19/sql-server-alwayson-availability-groups-supported-between-microsoft-azure-regions.aspx> .

### Summary on SQL Server High Availability in Azure

Given the fact that Azure Storage is protecting the content, there is one less reason to insist on a hot-standby image. This means your High Availability scenario needs to only protect against the following cases:

* Unavailability of the VM as a whole due to maintenance on the server cluster in Azure or other reasons
* Software issues in the SQL Server instance
* Protecting from manual error where data gets deleted and point-in-time recovery is needed

Looking at matching technologies one can argue that the first two cases can be covered by Database Mirroring or AlwaysOn, whereas the third case only can be covered by Log-Shipping.

You will need to balance the more complex setup of AlwaysOn, compared to Database Mirroring, with the advantages of AlwaysOn. Those advantages can be listed like:

* Readable secondary replicas.
* Backups from secondary replicas.
* Better scalability.
* More than one secondary replicas.

## General SQL Server for SAP on Azure Summary

There are many recommendations in this guide and we recommend you read it more than once before planning your Azure deployment. In general, though, be sure to follow the top ten general DBMS on Azure specific points:

1. Use the latest DBMS release, like SQL Server 2014, that has the most advantages in Azure.
   1. For SQL Server, this is SQL Server 2012 SP1 CU4 which would include the feature of backing up against Azure Storage. However, in conjunction with SAP we would recommend at least SQL Server 2014 SP1 CU1 or SQL Server 2012 SP2 and the latest CU.
2. Carefully plan your SAP system landscape in Azure to balance the data file layout and Azure restrictions:
   1. Don’t have too many VHDs, but have enough to ensure you can reach your required IOPS.
   2. Remember that IOPS are also limited per Azure Storage Account and that Storage Accounts are limited within each Azure subscription (<https://azure.microsoft.com/en-us/documentation/articles/azure-subscription-service-limits/>).
   3. Only stripe across VHDs if you need to achieve a higher IOP throughput.
3. Never install software or put any files that require persistence on the D:\ drive as it is non-permanent and anything on this drive will be lost at a Windows reboot.
4. Don’t use Azure VHD caching for Azure Standard Storage.
5. Don’t use Azure geo-replicated Storage Accounts. Use Locally Redundant for DBMS workloads.
6. Use your DBMS vendor’s HA/DR solution to replicate database data.
7. Always use Name Resolution, don’t rely on IP addresses.
8. Use the highest database compression possible.
   1. For SQL Server this is page compression.
9. Be careful using Azure Gallery SQL Server images.
   1. If you use the SQL Server one, you must change the instance collation before installing any SAP NetWeaver system on it.
10. Install and configure the SAP Host Monitoring for Azure as described in ‘*SAP NetWeaver on Microsoft Azure Virtual Machine Service - Deployment Guide*’.

# Specifics to SAP Adaptive Server (SAP ASE) RDBMS

Starting with Microsoft Azure, you can easily migrate your existing SAP ASE applications to Azure Virtual Machines. SAP ASE in a Virtual Machine enables you to reduce the total cost of ownership of deployment, management and maintenance of enterprise breadth applications by easily migrating these applications to Microsoft Azure. With SAP ASE in an Azure Virtual Machine, administrators and developers can still use the same development and administration tools that are available on-premises.

There is an SLA for the Azure Virtual Machines which can be found here:

<http://www.windowsazure.com/en-us/support/legal/sla>

We are confident that Microsoft Azure hosted Virtual Machines will perform very well in comparison to other public cloud virtualization offerings, but individual results may vary. SAP sizing SAPS numbers of the different SAP certified VM SKUs will be provided in a separate SAP Note [1928533 – SAP Applications on Azure: Supported Products and Sizing](http://service.sap.com/sap/support/notes/1928533).

Statements and recommendations in regard to the usage of Azure Storage, Deployment of SAP VMs or SAP Monitoring apply to deployments of SAP ASE in conjunction with SAP applications as stated throughout chapters 1 to 3 of this document.

## SAP ASE Version Support

SAP currently supports SAP ASE version 16.0 for use with SAP Business Suite products. All updates for SAP ASE server, or JDBC and ODBC drivers to be used with SAP Business Suite products are provided solely through the SAP Service Marketplace at: <http://service.sap.com/swdc>.

As for installations on-premises, do not download updates for the SAP ASE server, or for the JDBC and ODBC drivers directly from Sybase websites. For detailed information on patches which are supported for use with SAP Business Suite products on-premises and in Azure Virtual Machines see the following SAP Notes:

[1590719 – SYB: Updates for SAP Adaptive Server Enterprise (SAP ASE)](https://service.sap.com/sap/support/notes/1590719)

[1973241 – SYB: Release information for SAP ASE 16.0 SP00 with SAP Business Suite](https://service.sap.com/sap/support/notes/1973241)

General information on running SAP Business Suite on SAP ASE can be found in the SCN: http://scn.sap.com/community/ase

## SAP ASE Configuration Guidelines for SAP related SAP ASE Installations in Azure VMs

### Structure of the SAP ASE Deployment

In accordance with the general description, SAP ASE executables should be located or installed into the system drive of the VM’s base VHD (drive c:\). Typically, most of the SAP ASE system and tools databases are not really leveraged hard by SAP NetWeaver workload. Hence the system and tools databases (master, model, saptools, sybmgmtdb, sybsystemdb) can remain on the C:\drive as well.

An exception could be the temporary database containing all work tables and temporary tables created by SAP ASE, which in case of some SAP ERP and all BW workloads might require either higher data volume or I/O operations volume which can’t fit into the original VM’s base VHD (drive c:\).

Depending on the version of SAPInst/SWPM used to install the system, the database might contain:

* A single SAP ASE tempdb which is created when installing SAP ASE
* An SAP ASE tempdb created by installing SAP ASE and an additional saptempdb created by the SAP installation routine
* An SAP ASE tempdb created by installing SAP ASE and an additional tempdb that has been created manually (e.g. following SAP Note [1752266 – SYB: How to create an additional tempdb with Sybase ASE) to meet ERP/BW specific tempdb requirements](https://service.sap.com/sap/support/notes/1752266)

In case of specific ERP or all BW workloads it makes sense, in regard to performance, to keep the tempdb devices of the additionally created tempdb (by SWPM or manually) on a drive other than C:\. If no additional tempdb exists it is recommended to create one (SAP Note [1752266](https://service.sap.com/sap/support/notes/1752266)).

For such systems the following steps should be performed for the additionally created tempdb:

* Move the first tempdb device to the first device of the SAP database
* Add tempdb devices to each of the VHDs containing a devices of the SAP database

This configuration enables tempdb to either consume more space than the system drive is able to provide. As a reference one can check the tempdb device sizes on existing systems which run on-premises. Or such a configuration would enable IOPS numbers against tempdb which cannot be provided with the system drive. Again systems that are running on-premises can be used to monitor I/O workload against tempdb.

Never put any SAP ASE devices onto the D:\ drive of the VM. This also applies to the tempdb, even if the objects kept in the tempdb are only temporary.

### Impact of Database Compression

In configurations where I/O bandwidth can become a limiting factor, every measure which reduces IOPS might help to stretch the workload one can run in an IaaS scenario like Azure. Therefore, it is strongly recommended to make sure that SAP ASE compression is used before uploading an existing SAP database to Azure.

The recommendation to perform compression before uploading to Azure if it is not already implemented is given out of several reasons:

* The amount of data to be uploaded to Azure is lower
* The duration of the compression execution is shorter assuming that one can use stronger hardware with more CPUs or higher I/O bandwidth or less I/O latency on-premises
* Smaller database sizes might lead to less costs for disk allocation

Data- and LOB-Compression work as well functionally in a VM hosted in Azure Virtual Machine Services as it does on-premises. For more details on how to check if compression is already in use in an existing SAP ASE database please check SAP Note [1750510](https://service.sap.com/sap/support/notes/1750510) – Sybase ASE 15.7 Compression.

### Using DBACockpit to monitor Database Instances

For SAP systems which are using SAP ASE as database platform, the DBACockpit is accessible as embedded browser windows in transaction DBACockpit or as Webdynpro. However the full functionality for monitoring and administering the database is available in the Webdynpro implementation of the DBACockpit only.

As with on-premises systems several steps are required to enable all SAP NetWeaver functionality used by the Webdynpro implementation of the DBACockpit. Please follow SAP Note [1245200 - SYB: DBA Cockpit available for Sybase ASE](https://service.sap.com/sap/support/notes/1245200) to enable the usage of webdynpros and generate the required ones. When following the instructions in the above notes you will also configure the Internet Communication Manager (icm) along with the ports to be used for http and https connections. The default setting for http looks like this:

icm/server\_port\_0 = PROT=HTTP,PORT=8000,PROCTIMEOUT=600,TIMEOUT=600  
icm/server\_port\_1 = PROT=HTTPS,PORT=443$$,PROCTIMEOUT=600,TIMEOUT=600

and the links generated in transaction DBACockpit will look similar to this:

https://<fullyqualifiedhostname>:44300/sap/bc/webdynpro/sap/dba\_cockpit

http://<fullyqualifiedhostname>:8000/sap/bc/webdynpro/sap/dba\_cockpit

Depending on if and how the Azure Virtual Machine hosting the SAP system is connected via site-to-site, multi-site or ExpressRoute (Cross-Premises deployment) you need to make sure that ICM is using a fully qualified hostname that can be resolved on the machine where you are trying to open the DBACockpit from. Please see SAP Note [773830 – FQHN determination in ICM](https://service.sap.com/sap/support/notes/773830) to understand how ICM determines the fully qualified host name depending on profile parameters and set parameter icm/host\_name\_full explicitly if required.

If you deployed the VM in a Cloud-Only scenario without cross-premises connectivity between on-premises and Azure, you need to use the DNS name of the Azure Cloud Service. Setting the SAP profile parameter icm/host\_name\_full to the DNS name of the Azure Cloud Service the link might look similar to:

<https://myazurevm.cloudapp.net:44300/sap/bc/webdynpro/sap/dba_cockpit>

<http://myazurevm.cloudapp.net:8000/sap/bc/webdynpro/sap/dba_cockpit>

In this case you need to make sure to:

* Add endpoints in the Azure Portal for the TCP/IP ports used to communicate with ICM
* Add Inbound rules to the Windows Firewall configuration for the TCP/IP ports used to communicate with the ICM

For an automated imported of all corrections available, it is recommended to periodically apply the correction collection SAP Note applicable to your SAP version:

[1558958 - DBA Cockpit Patch Collection 7.02/7.30](https://service.sap.com/sap/support/notes/1558958)

[1619967 - DBA Cockpit Patch Collection 7.31](https://service.sap.com/sap/support/notes/1619967)

[1882376 - DBA Cockpit Patch Collection 7.40](https://service.sap.com/sap/support/notes/1882376)

Further information about DBA Cockpit for SAP ASE can be found in the following SAP Notes:

[1605680 - SYB: Troubleshoot the setup of the DBA Cockpit on Sybase ASE](https://service.sap.com/sap/support/notes/1605680)

[1757924 - Release Notes 7.02 SP10, 7.30 SP5, 7.31 SP1](https://service.sap.com/sap/support/notes/1757924)

[1757928 - Release Notes 7.02 SP11, 7.30 SP6, 7.31 SP2](https://service.sap.com/sap/support/notes/1757928)

[1758182 - Release Notes 7.02 SP12, 7.30 SP8, 7.31 SP5](https://service.sap.com/sap/support/notes/1758182)

[1758496 - Release Notes 7.02 SP13, 7.30 SP9, 7.31 SP7, 7.40 SP2](https://service.sap.com/sap/support/notes/1758496)

[1814258 - Release Notes 7.02 SP14, 7.30 SP10, 7.31 SP9, 7.40 SP4](https://service.sap.com/sap/support/notes/1814258)

[1922555 - Release Notes 7.02 SP15, 7.30 SP11, 7.31 SP11, 7.40 SP6](https://service.sap.com/sap/support/notes/1922555)

[1956005 - Release Notes 7.02 SP16, 7.30 SP12, 7.31 SP13, 7.40 SP8](https://service.sap.com/sap/support/notes/1956005)

### Backup/Recovery Considerations for SAP ASE

When deploying SAP ASE into Azure your backup methodology must be reviewed. Even if the system is not a productive system, the SAP database hosted by SAP ASE must be backed up periodically. Since Azure Storage keeps three images, a backup is now less important in respect to compensating a storage crash. The priority reason for maintaining a proper backup and restore plan is more that you can compensate for logical/manual errors by providing point in time recovery capabilities. So the goal is to either use backups to restore the database back to a certain point in time or to use the backups in Azure to seed another system by copying the existing database. For example, you could transfer from a 2-Tier SAP configuration to a 3-Tier system setup of the same system by restoring a backup.

Backing up and restoring a database in Azure works the same way as it does on-premises. Please see SAP Notes:

[1588316 – SYB: Configure automatic database and log backups](https://service.sap.com/sap/support/notes/1588316)

[1585981 – SYB: Ensuring Recoverability for SAP ASE](https://service.sap.com/sap/support/notes/1585981)

for details on creating dump configurations and scheduling backups. Depending on your strategy and needs you can configure database and log dumps to disk onto one of the existing VHDs or add an additional VHD for the backup. To reduce the danger of data loss in case of an error it is recommended to use a VHD where no database device is located.

Besides data- and LOB compression SAP ASE also offers backup compression. To occupy less space with the database and log dumps it is recommended to use backup compression. Please see SAP Note [1588316 – SYB: Configure automatic database and log backups](https://service.sap.com/sap/support/notes/1588316) for more information. Compressing the backup is also crucial to reduce the amount of data to be transferred if you plan to download backups or VHDs containing backup dumps from the Azure Virtual Machine to on-premises.

Do not use drive D:\ as database or log dump destination.

### Performance Considerations for Backups/Restores

As in bare-metal deployments, backup/restore performance is dependent on how many volumes can be read in parallel and what the throughput of those volumes might be. In addition, the CPU consumption used by backup compression may play a significant role on VMs with just up to 8 CPU threads. Therefore, one can assume:

* The fewer the number of VHDs used to store the database devices, the smaller the overall throughput in reading
* The smaller the number of CPU threads in the VM, the more severe the impact of backup compression
* The fewer targets (Stripe Directories, VHDs) to write the backup to, the lesser the throughput

To increase the number of targets to write to there are two options which can be used/combined depending on your needs:

* Striping the backup target volume over multiple mounted VHDs in order to improve the IOPS throughput on that striped volume
* Creating a dump configuration on SAP ASE level which uses more than one target directory to write the dump to

Striping a volume over multiple mounted VHDs has been discussed earlier in this guide. For more information on using multiple directories in the SAP ASE dump configuration please refer to the documentation on Stored Procedure sp\_config\_dump which is used to create the dump configuration:

<http://infocenter.sybase.com/help/index.jsp?topic=/com.sybase.infocenter.dc36273.1600/doc/html/san1393051993446.html>

## Disaster Recovery with Azure VMs

### Data Replication with SAP Sybase Replication Server

With the SAP Sybase Replication Server (SRS) SAP ASE provides a warm standby solution to transfer database transactions to a distant location asynchronously.

The installation and operation of SRS works as well functionally in a VM hosted in Azure Virtual Machine Services as it does on-premises.

ASE HADR via SAP Replication Server is planned with a future release. It will be tested with and released for Windows Azure platforms as soon as it is available.

# Specifics to Oracle Database

Since midyear 2013, Oracle software is supported by Oracle to run on Microsoft Windows Hyper-V and Azure. Please read this article to get more details on the general support of Windows Hyper-V and Azure by Oracle: <https://blogs.oracle.com/cloud/entry/oracle_and_microsoft_join_forces>

Following the general support, the specific scenario of SAP applications leveraging Oracle Databases is supported as well. Details are named in this part of the document.

## Oracle Version Support

All details about Oracle versions and corresponding OS versions which are supported for running SAP on Oracle on Azure Virtual Machines can be found in the following SAP Note:  
[2039619 - SAP Applications on Azure using Oracle Database: Supported Products and Versions](https://service.sap.com/sap/support/notes/2039619)

General information about running SAP Business Suite on Oracle can be found on SCN: <http://scn.sap.com/community/oracle>

## Oracle Configuration Guidelines for SAP Installations in Azure VMs

### **Storage configuration**

Only single instance Oracle using NTFS formatted disks is supported. All database files must be stored on the NTFS file system based on VHD disks. These VHDs are mounted to the Azure VM and are based on Azure Page BLOB Storage (<http://msdn.microsoft.com/en-us/library/azure/ee691964.aspx> ).   
Any kind of network drives or remote shares like Azure file services:

* <http://blogs.msdn.com/b/windowsazurestorage/archive/2014/05/12/introducing-microsoft-azure-file-service.aspx>
* <http://blogs.msdn.com/b/windowsazurestorage/archive/2014/05/27/persisting-connections-to-microsoft-azure-files.aspx>

are **NOT** supported for Oracle database files!

Using Azure VHDs based on Azure Page BLOB Storage, the statements made in this document in [chapter 1.4](#_Caching_for_VMs) and [chapter 1.6](#_Microsoft_Azure_Storage) apply to deployments with the Oracle Database as well.

As explained earlier in the general part of the document, quotas on IOPS throughput for Azure VHDs exist. The exact quotas are depending on the VM type used. A list of VM types with their quotas can be found here: <http://msdn.microsoft.com/library/azure/dn197896.aspx>

To identify the supported Azure VM types, please refer to SAP note [1928533 – SAP Applications on Azure: Supported Products and Sizing](http://service.sap.com/sap/support/notes/1928533)

As long as the current IOPS quota per disk satisfies the requirements, it is possible to store all the DB files on one single mounted Azure VHD.

If more IOPS are required, it is strongly recommended to use Window Storage Pools (only available in Windows Server 2012 and higher) or Windows striping for Windows 2008 R2 to create one big logical device over multiple mounted VHD disks. See also [chapter 1.5](#_Striping_and_Storage) of this document. This approach simplifies the administration overhead to manage the disk space and avoids the effort to manually distribute files across multiple mounted VHDs.

### Backup / Restore

For backup / restore functionality, the SAP BR\*Tools for Oracle are supported in the same way as on standard Windows Server Operating Systems and Hyper-V. Oracle Recovery Manager (RMAN) is also supported for backups to disk and restore from disk.

### High Availability

Oracle Data Guard is supported for high availability and disaster recovery purposes. Details can be found  
in this documentation: <http://msdn.microsoft.com/en-us/library/azure/dn614680.aspx>

### Other

All other general topics like Azure Availability Sets or SAP monitoring apply as described in chapters 1 through 3 of this document for deployments of VMs with the Oracle Database as well.

# Specifics for the SAP MaxDB Database

## SAP MaxDB Version Support

SAP currently supports **SAP MaxDB** **version 7.9** for use with SAP NetWeaver-based products in Azure. All updates for SAP MaxDB server, or JDBC and ODBC drivers to be used with SAP NetWeaver-based products are provided solely through the SAP Service Marketplace at <http://support.sap.com/swdc>.

General information on running SAP NetWeaver on SAP MaxDB can be found at <http://scn.sap.com/community/maxdb>.

## Supported Microsoft Windows Versions and Azure VM types for SAP MaxDB DBMS

To find the supported Microsoft Windows version for SAP MaxDB DBMS on Azure, see:

* SAP Product Availability Matrix (PAM): <https://support.sap.com/pam>
* [SAP Note 1928533 - SAP Applications on Azure: Supported Products and Azure VM types](http://service.sap.com/sap/support/notes/1928533)

It is highly recommended to use the newest version of the operating system Microsoft Windows, which is Microsoft Windows 2012 R2.

## Available SAP MaxDB Documentation

You can find the updated list of SAP MaxDB documentation in the following SAP Note:

[767598 - Available SAP MaxDB documentation](https://service.sap.com/sap/support/notes/767598)

## SAP MaxDB Configuration Guidelines for SAP Installations in Azure VMs

### **Storage configuration**

Azure storage best practices for SAP MaxDB follow the general recommendations mentioned in chapter [2 Structure of an RDBMS Deployment](#_Structure_of_an).

Achtung **Important**

Like other databases, SAP MaxDB also has data and log files. However, in SAP MaxDB terminology the correct term is “volume” (not “file”). For example, there are SAP MaxDB data volumes and log volumes. Do not confuse these with OS disk volumes.

In short you have to:

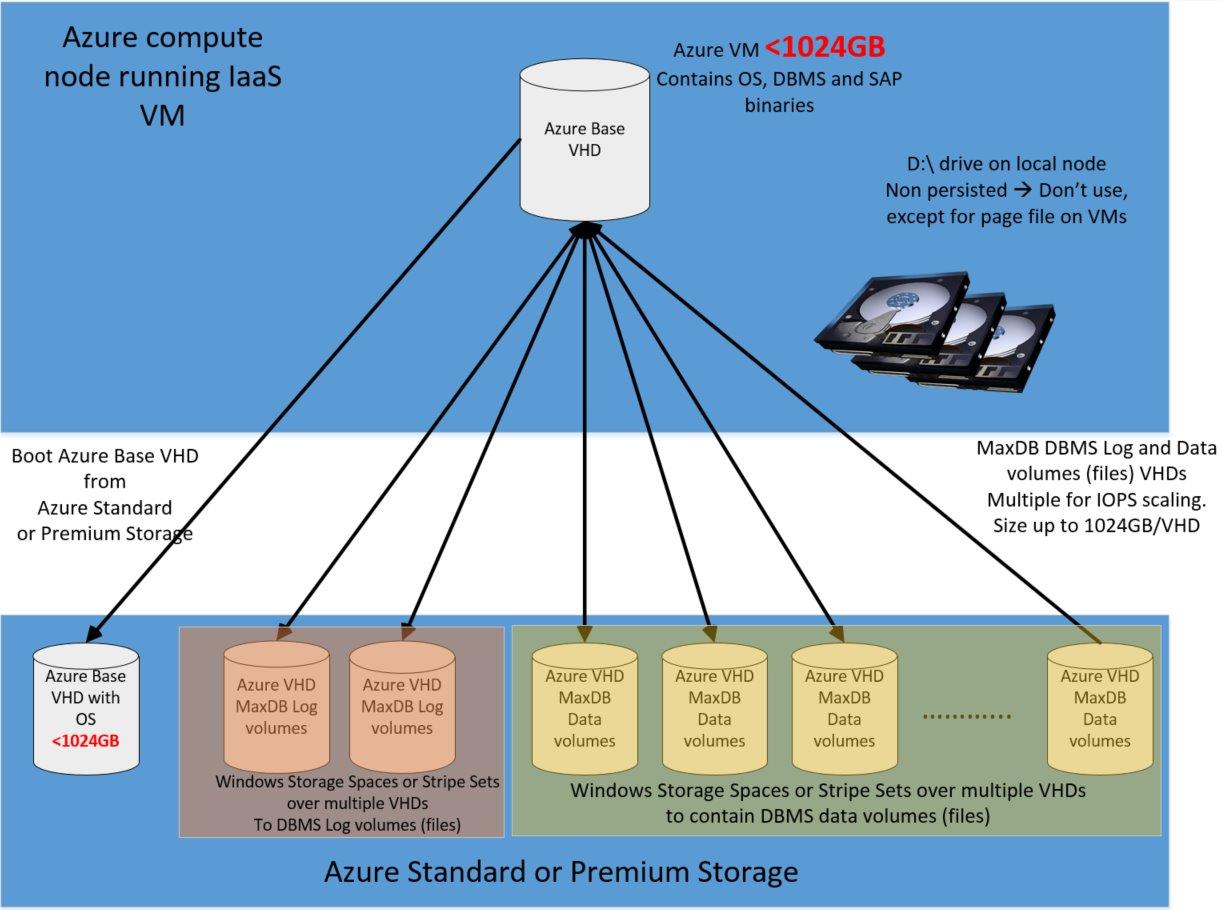
* Set the Azure storage account that holds the SAP MaxDB data and log volumes (i.e. files) to **Local Redundant Storage (LRS)** as specified in chapter [2.3 Microsoft Azure Storage](#_Microsoft_Azure_Storage).
* Separate the IO path for SAP MaxDB data volumes (i.e. files) from the IO path for log volumes (i.e. files). This means that SAP MaxDB data volumes (i.e. files) have to be installed on one logical drive and SAP MaxDB log volumes (i.e. files) have to be installed on another logical drive.
* Set the proper file caching for each Azure blob, depending on whether you use it for SAP MaxDB data or log volumes (i.e. files), and whether you use Azure Standard or Azure Premium Storage, as described in [chapter 2.1 Caching for VMs](#_Caching_for_VMs).
* As long as the current IOPS quota per disk satisfies the requirements, it is possible to store all the data volumes on a single mounted Azure VHD, and also store all database log volumes on another single mounted Azure VHD.
* If more IOPS and/or space are required, it is strongly recommended to use Microsoft Window Storage Pools (only available in Microsoft Windows Server 2012 and higher) or Microsoft Windows striping for Microsoft Windows 2008 R2 to create one large logical device over multiple mounted VHD disks. See also [chapter 2.2 Striping and Storage Spaces](#_Striping_and_Storage) of this document. This approach simplifies the administration overhead to manage the disk space and avoids the effort of manually distributing files across multiple mounted VHDs.
* For the highest IOPS requirements, you can use Azure Premium Storage, which is available on DS-series and GS-series VMs.

Figure 6 : Reference Configuration of Azure IaaS VM for SAP MaxDB DBMS

### Backup and Restore

When deploying SAP MaxDB into Azure, you must review your backup methodology. Even if the system is not a productive system, the SAP database hosted by SAP MaxDB must be backed up periodically. Since Azure Storage keeps three images, a backup is now less important in terms of protecting your system against storage failure and more important operational or administrative failures. The primary reason for maintaining a proper backup and restore plan is so that you can compensate for logical or manual errors by providing point-in-time recovery capabilities. So the goal is to either use backups to restore the database to a certain point in time or to use the backups in Azure to seed another system by copying the existing database. For example, you could transfer from a 2-tier SAP configuration to a 3-tier system setup of the same system by restoring a backup.

Backing up and restoring a database in Azure works the same way as it does for on-premises systems, so you can use standard SAP MaxDB backup/restore tools, which are described in one of the SAP MaxDB documentation documents listed in SAP Note [767598 - Available SAP MaxDB documentation](https://service.sap.com/sap/support/notes/767598).

### Performance Considerations for Backup and Restore

As in bare-metal deployments, backup and restore performance is dependent on how many volumes can be read in parallel and the throughput of those volumes. In addition, the CPU consumption used by backup compression can play a significant role on VMs with up to 8 CPU threads. Therefore, one can assume:

* The fewer the number of VHDs used to store the database devices, the lower the overall read throughput
* The smaller the number of CPU threads in the VM, the more severe the impact of backup compression
* The fewer targets (Stripe Directories, VHDs) to write the backup to, the lower the throughput

To increase the number of targets to write to, there are two options that you can use, possibly in combination, depending on your needs:

* Dedicating separate volumes for backup
* Striping the backup target volume over multiple mounted VHDs in order to improve the IOPS throughput on that striped disk volume
* Having separate dedicated logical disk devices for:
  + SAP MaxDB backup volumes (i.e. files)
  + SAP MaxDB data volumes ( i.e. files)
  + SAP MaxDB log volumes ( i.e. files)

Striping a volume over multiple mounted VHDs has been discussed earlier in [chapter 2.2 Striping and Storage Spaces](#_Striping_and_Storage) of this document.

### Other

All other general topics such as Azure Availability Sets or SAP monitoring also apply as described in chapters 1 through 3 of this document for deployments of VMs with the SAP MaxDB database.

Other SAP MaxDB-specific settings are transparent to Azure VMs and are described in different documents listed in SAP Note [767598 - Available SAP MaxDB documentation](https://service.sap.com/sap/support/notes/767598) and in these SAP notes:

[826037 - FAQ: SAP MaxDB/liveCache Support](http://service.sap.com/sap/support/notes/826037)

[1139904 - FAQ: SAP MaxDB/liveCache database parameters](http://service.sap.com/sap/support/notes/1139904)

[1173395 - FAQ: SAP MaxDB and liveCache configuration](http://service.sap.com/sap/support/notes/1173395)

# Specifics for SAP liveCache

## SAP liveCache Version Support

Minimal version of SAP liveCache supported in Azure Virtual Machines is: **SAP LC/LCAPPS 10.0 SP 25** including **liveCache 7.9.08.31** and **LCA-Build 25**, released **for EhP 2 for SAP SCM 7.0** and higher.

## Supported Microsoft Windows Versions and Azure VM types for SAP liveCache DBMS

To find the supported Microsoft Windows version for SAP liveCache on Azure, see:

* SAP Product Availability Matrix (PAM): <https://support.sap.com/pam>
* [SAP Note 1928533 - SAP Applications on Azure: Supported Products and Azure VM types](http://service.sap.com/sap/support/notes/1928533)

It is highly recommended to use the newest version of the operating system Microsoft Windows, which is Microsoft Windows 2012 R2.

## SAP liveCache Configuration Guidelines for SAP Installations in Azure VMs

### Recommended Azure VM Types

As SAP liveCache is an application that performs huge calculations, the amount and speed of RAM and CPU has a major influence on SAP liveCache performance.

For the Azure VM types supported by SAP ([SAP Note 1928533 - SAP Applications on Azure: Supported Products and Azure VM types](http://service.sap.com/sap/support/notes/1928533)), all virtual CPU resources allocated to the VM are backed by dedicated physical CPU resources of the hypervisor. No overprovisioning (and therefore no competition for CPU resources) takes place.

Similarly, for all Azure VM instance types supported by SAP, the VM memory is 100 % mapped to the physical memory – overprovisioning (over-commitment), for example, is not used.

From this perspective it is highly recommended to use the new D- series or DS-series (in combination with Azure Premium Storage) Azure VM type, as they have 60 % faster processors than the A-series. For the highest RAM and CPU load, you can use G-series and GS-series (in combination with Azure Premium Storage) VMs with the latest [Intel® Xeon® processor E5 v3 family](http://www.intel.com/content/www/us/en/processors/xeon/xeon-e5-solutions.html), which have twice the memory and four times the solid state drive storage (SSDs) of the D/DS-series.

### Storage Configuration

As SAP liveCache is based on SAP MaxDB technology, all the Azure storage best practice recommendations mentioned for SAP MaxDB in [chapter 8.4.1 Storage configuration](#_Storage_configuration) are also valid for SAP liveCache.

### Dedicated Azure VM for liveCache

As SAP liveCache intensively uses computational power, for productive usage it is highly recommended to deploy on a dedicated Azure Virtual Machine.

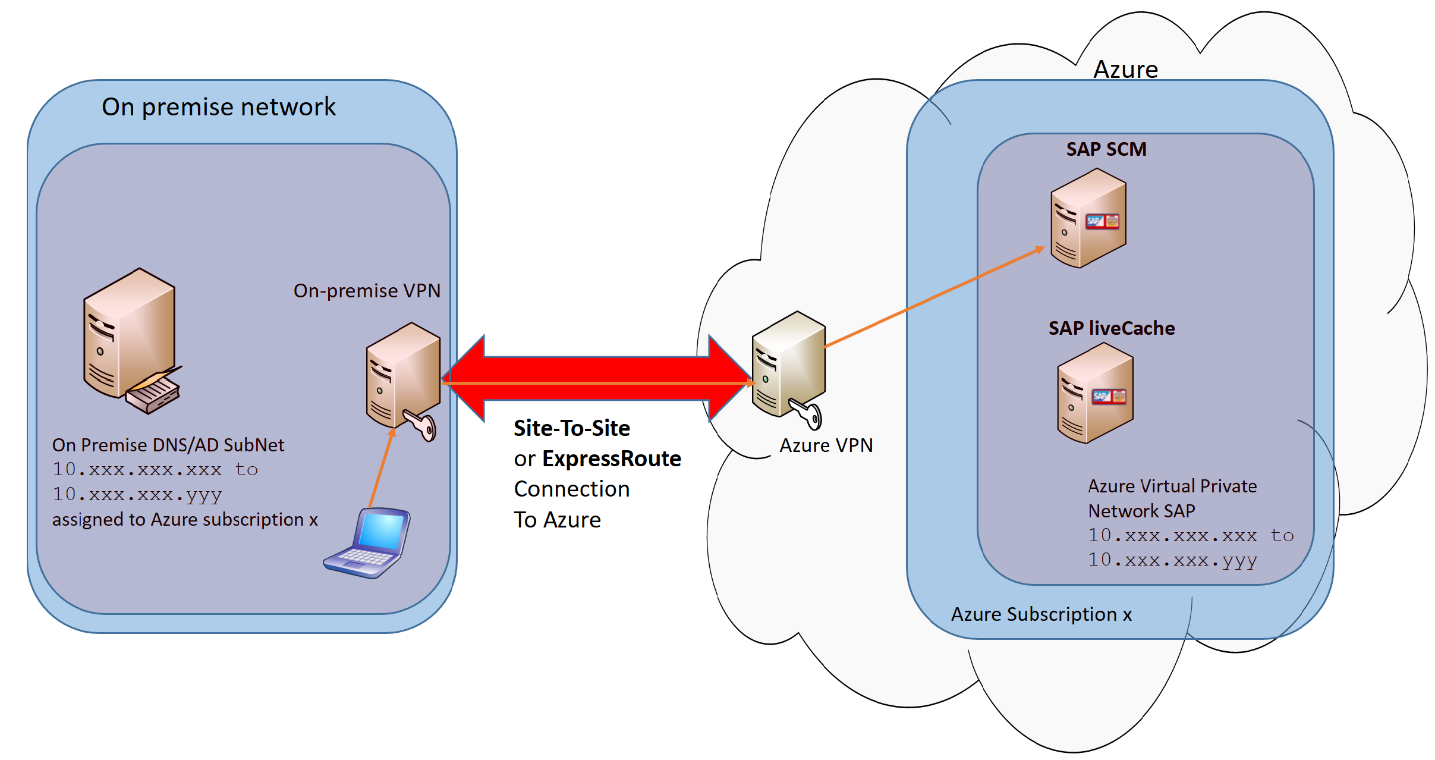


Figure 7: Dedicated Azure VM for liveCache for productive use case

### Backup and Restore

Backup and restore, including performance considerations, are already described in the relevant SAP MaxDB chapters [8.4.2 Backup and Restore](#_Backup_/_Restore) and [8.4.3 Performance Considerations for Backup and Restore](#_Performance_Considerations_for).

### Other

All other general topics are already described in the relevant SAP MaxDB chapter [9.3.5 Other](#_Other).

# Specifics for the SAP Content Server

The SAP Content Server is a separate, server-based component to store content such as electronic documents in different formats. The SAP Content Server is provided by development of technology and is to be used cross-application for any SAP applications. It is installed on a separate system. Typical content is training material and documentation from Knowledge Warehouse or technical drawings originating from the mySAP PLM Document Management System.

## SAP Content Server Version Support

SAP currently supports:

* **SAP Content Server** with version **6.50 (and higher)**
* You have to use **SAP MaxDB** **version 7.9**
* **Microsoft IIS (Internet Information Server) version 8.0 (and higher)**

It is highly recommended to use the newest version of SAP Content Server, which at the time of writing this document is **6.50 SP4**, and the newest version of **Microsoft IIS 8.5**.

Check the latest supported versions of SAP Content Server and Microsoft IIS in the SAP Product Availability Matrix (PAM) at <https://service.sap.com/pam>.

## Supported Microsoft Windows and Azure VM types for SAP Content Server

To find out supported Windows version for SAP Content Server on Azure, see:

* SAP Product Availability Matrix (PAM): <https://service.sap.com/pam>
* [SAP Note 1928533 - SAP Applications on Azure: Supported Products and Azure VM types](http://service.sap.com/sap/support/notes/1928533)

It is highly recommended to use the newest version of Microsoft Windows, which at the time of writing this document is **Windows 2012 R2**.

## SAP Content Server Configuration Guidelines for SAP Installations in Azure VMs

### Storage Configuration

If you configure SAP Content Server to store files in the SAP MaxDB database, all Azure storage best practices recommendation mentioned for SAP MaxDB in [chapter 8.4.1 Storage Configuration](#_Storage_configuration) are also valid for the SAP Content Server scenario.

If you configure SAP Content Server to store files in the file system, it is recommended to use a dedicated logical drive. Using storage spaces enables you to also increase logical disk size and IOPS throughput, as described in in [chapter 2.2 Striping and Storage Spaces](#_Striping_and_Storage).

### SAP Content Server Location

SAP Content Server has to be deployed in the same Azure region and Azure VNET where the SAP system is deployed. You are free to decide whether you want to deploy SAP Content Server components on a dedicated Azure VM or on the same VM where the SAP system is running.

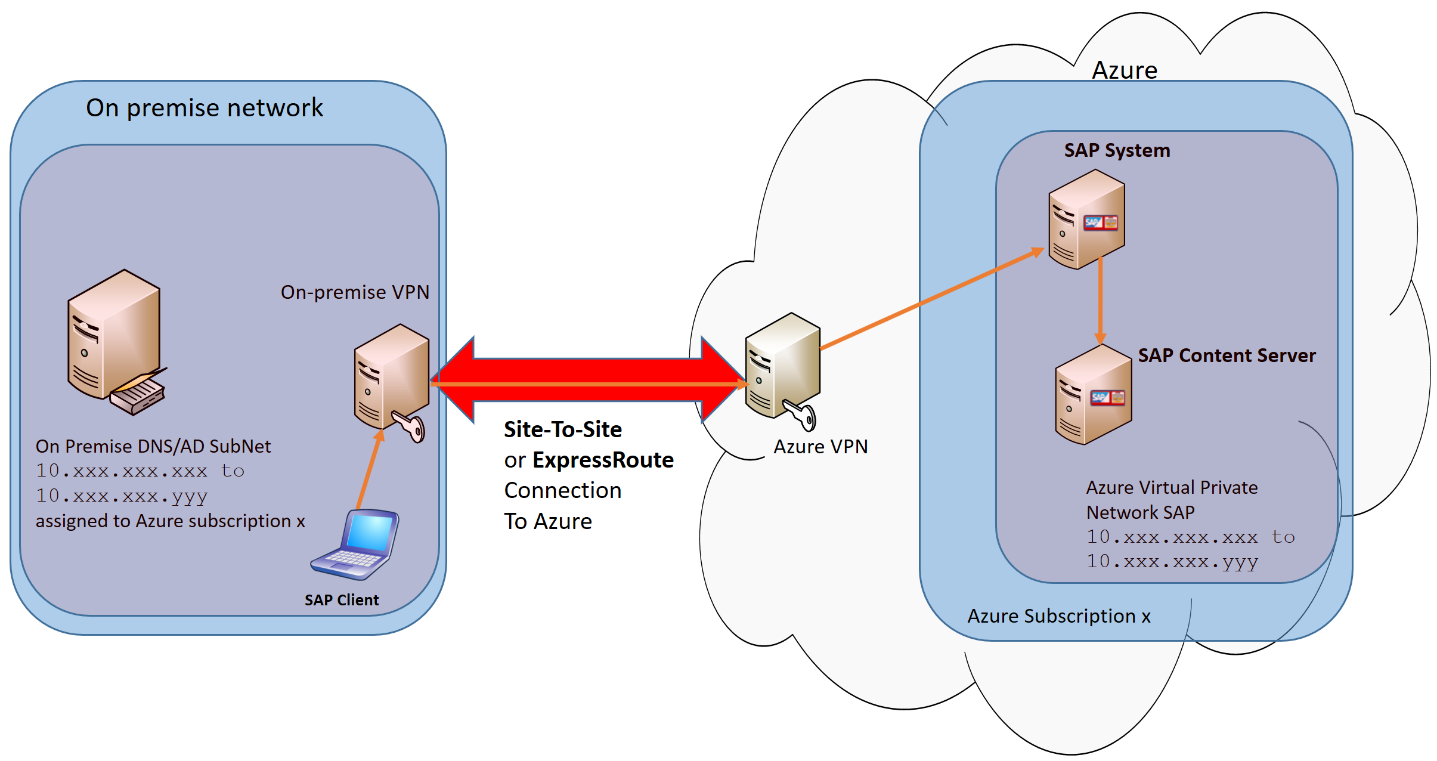


Figure 8 Dedicated Azure VM for SAP Content Server

### SAP Cache Server Location

The SAP Cache Server is an additional server-based component to provide access to (cached) documents locally. The SAP Cache Server caches the documents of a SAP Content Server. This is to optimize network traffic if documents have to be retrieved more than once from different locations. The general rule is that the SAP Cache Server has to be physically close to the client that accesses the SAP Cache Server.

Here you have two options:

1. **Client is a backend SAP system**

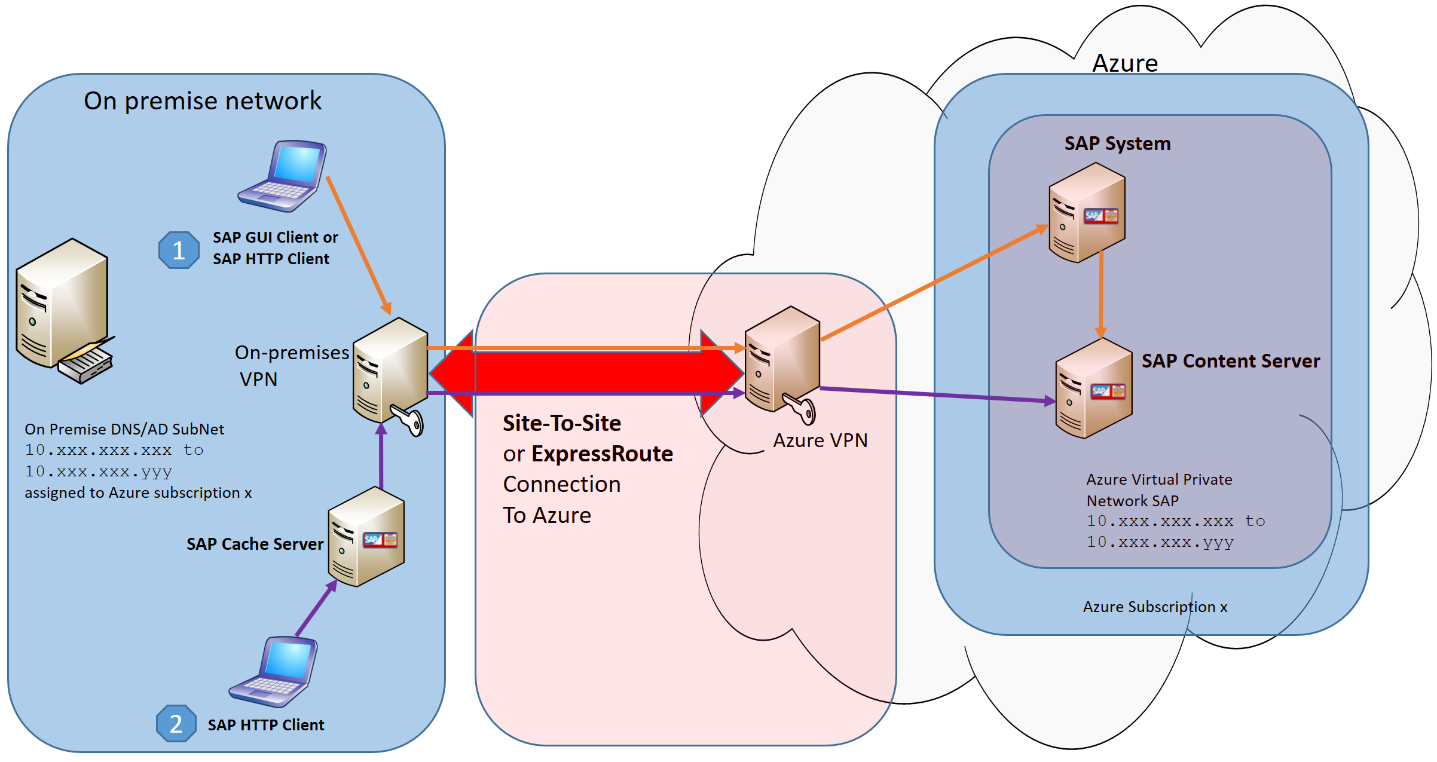
If a backend SAP system is configured to access SAP Content Server, that SAP system is a client. As both SAP system and SAP Content Server are deployed in the same Azure region – in the same Azure datacenter – they are physically close to each other. Therefore, there is no need to have a dedicated SAP Cache Server. SAP UI clients (SAP GUI or web browser) access the SAP system directly, and the SAP system retrieves documents from the SAP Content Server.

1. **Client is an on-premises web browser**

The SAP Content Server can be configured to be accessed directly by the web browser. In this case, a web browser running on–premise is a client of the SAP Content Server. On-premises datacenter and Azure datacenter are placed in different physical locations (ideally close to each other). Your on-premises datacenter is connection to Azure via Azure Site-to-Site VPN or ExpressRoute network connection. Although both options offer secure VPN network connection to Azure, site-to-site network connection does not offer network bandwidth and latency SLA between the on-premises datacenter and the Azure datacenter. To speed up access to documents, you can do one of the following:

**a)** Install SAP Cache Server on-premises, close to the on-premises web browser (option  on Figure 9)

**b)** Configure Azure ExpressRoute, which offers a high-speed and low-latency dedicated network connection between on-premises datacenter and Azure datacenter.



**Figure 9 Option to install SAP Cache Server on-premises**

### Backup / Restore

If you configure the SAP Content Server to store files in the SAP MaxDB database, the backup/restore procedure and performance considerations are already described in SAP MaxDB [chapter 8.4.2 Backup / Restore](#_Backup_/_Restore) and chapter [8.4.3 Performance Considerations for Backups/Restores](#_Performance_Considerations_for).

If you configure the SAP Content Server to store files in the file system, one option is to execute manual backup/restore of the whole file structure where the documents are located. Similar to SAP MaxDB backup/restore, it is recommended to have a dedicated disk volume for backup purpose.

### Other

Other SAP Content Server specific settings are transparent to Azure VMs and are described in various documents and SAP Notes:

<https://service.sap.com/contentserver>

[1619726 - FAQ: SAP MaxDB Content Server](http://service.sap.com/sap/support/notes/1619726)

<http://help.sap.com/> -> Technology Platform -> SAP Business Warehouse -> SAP NetWeaver BI Content   
-> SAP Content Server