



SQL Server Platform Choices

Module 9

Learning Units covered in this Module

- Lesson 1: SQL Server Editions
- Lesson 2: SQL Server on Linux
- Lesson 3: SQL Server on Containers
- Lesson 4: SQL Server in Microsoft Azure

Lesson 1: SQL Server Editions

Objectives

After completing this learning, you will be able to:

- Understand the editions of SQL Server 2022 available
- Understand how the editions differ from each other



SQL Server 2022 editions

Azure-enabled with continued performance and security innovation



Express

Free, entry-level database for small web and mobile apps

Feature highlights

- Up to 4 cores of CPU
- Up to 1410 MBs of memory
- Microsoft Purview Policies
- Azure AD authentication
- Built-in query intelligence: PSP Optimization, Optimized plan forcing
- Query store on by default for new databases
- Data Lake Virtualization
- Ledger
- Timeseries support



Standard

Full featured database for mid-tier applications and data marts

Feature highlights

- Up to 24 cores of CPU
- Up to 128 GBs of memory
- Azure Synapse Link for SQL
- Link feature for Azure SQL Managed Instance (basic availability groups)
- Buffer Pool Parallel Scan
- Backups to S3-compatible object storage

+ Express features



Enterprise

Mission-critical performance and intelligence for tier 1 databases

Feature highlights

- Unlimited cores of CPU
- Unlimited memory
- Azure Synapse Link for SQL (multi-threaded snapshot)
- Link feature for Azure SQL Managed Instance
- Built-in query intelligence: DOP feedback, CE Feedback, Memory Grant Feedback
- Contained Availability Group
- AVX 512 extension for batch mode

+ Standard features

+ Express features



Developer

Free to use with all the features of Enterprise Edition specifically for dev/test in non-production environments

← Build once and deploy across any SQL Server edition without changing your app →

SQL Server 2022 continues to deliver unparalleled value



Business continuity
through Azure



Seamless analytics over
on-prem operational data



Visibility over your
entire data estate

Most secure over
the last 10 years⁶



Industry-leading
performance and availability

2022 additions

- Azure extension for SQL Server
- Link feature for Azure SQL Managed Instance
- Azure Synapse link for SQL
- Azure Purview policies
- SQL Server Ledger
- Large memory and concurrency scalability
- Data virtualization for any data lake
- Object storage backup and restore
- Query Store on by default with replica support
- Query Store hints
- Intelligent Query Processing NextGen
- JSON functions
- Modern T-SQL surface area
- Time series support
- Integrated acceleration & offloading

2019 additions

- Azure Machine Learning and Spark ML
- Support for Kubernetes deployment
- Free supported Java
- Native UTF-8 support
- In-Memory Database: Persistent Memory support
- Accelerated database recovery
- Free DR to Azure Always Encrypted with secure enclaves
- Data classification & auditing
- Vulnerability assessment

2017 additions

- Support for Linux including Red Hat Enterprise Linux (RHEL), SUSE Linux Enterprise Server (SLES), and Ubuntu
- Support for Docker containers on Linux and Windows
- Python language support
- Support for graph data
- Automatic plan correction and adaptive query processing
- Cross platform availability groups

Knowledge Check

True or False: Only SQL Server 2022 can be directly connected to Azure Arc through the setup?

There are two new link options in SQL 2022. Which 2 are those?

Lesson 2: SQL Server on Linux

Objectives

After completing this learning, you will be able to:

- Understand the reasons to run SQL Server on Linux.
- Understand some of the SQL Server on Linux features.



SQL Server On Linux

Reasons To Run SQL Server On Linux

Linux Popularity

- Wide adoption in DevOps.
- A popular platform for Big Data Systems.
- Linux is popular platform for Cloud deployments.

Customer Demand

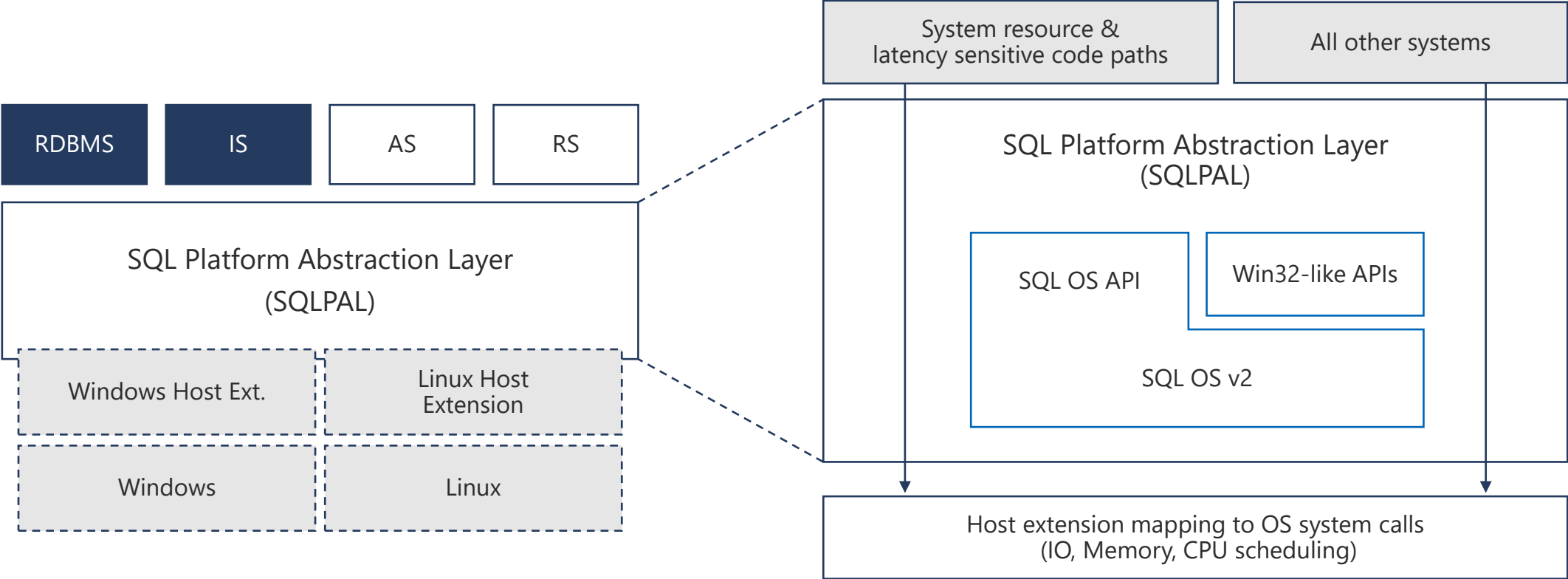
- Customers are happy with SQL Server but want options.
- Customers are looking for a lower cost of ownership.
- Customers are asking for support of multiple languages and frameworks to interact with SQL Server.

Microsoft's Commitment to Open Source

- Microsoft joined the Linux Foundation as a platinum member in November 2016.
- Microsoft is the top contributor on GitHub.
- Microsoft is a sponsor of Open Source Initiative.

SQL Server On Linux

Platform Abstraction Layer (PAL)



SQL Server 2022 On Linux

Supported Distributions

Red Hat Enterprise Linux – 8.x or 9.x Server

SUSE Enterprise Linux Server v15 (SP1 - SP4)

Ubuntu – 20.04 or 22.04

Docker Engine – 1.8+ on Linux

SQL Server On Linux

Unsupported Features

Notable unsupported features

- Stretch Database (deprecated in SQL 2022)
- SQL Agent subsystems: CmdExec, PowerShell, SSIS, SSAS, SSRS
- Analysis Services
- Reporting Services
- Data Quality Services
- Master Data Services
- SQL Server R Services
 - R is supported within SQL Server, but SQL Server R services as a separate package is not supported

SQL Server On Linux

Other Differences

SQL on Linux does not support named instances

- Use Containers instead

SQL Server is set to run under the mssql account which is a part of the mssql group.

- The mssql account permissions are set at install
- It is not supported to change ownership of the SQL Server binaries to another account

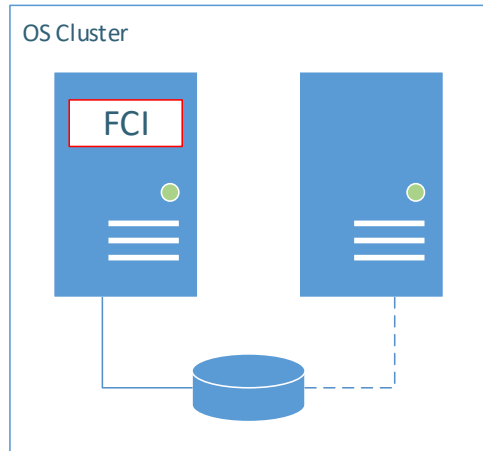
No SQL Server Browser Service

- Connections need to be done as ServerName if using port 1433 or ServerName, <port> if using a non-default port

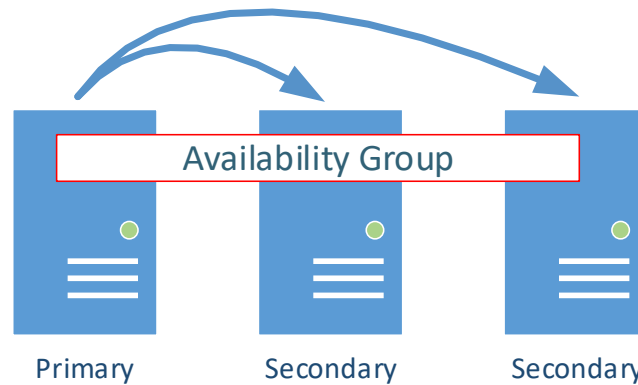
SQL Server On Linux

HADR

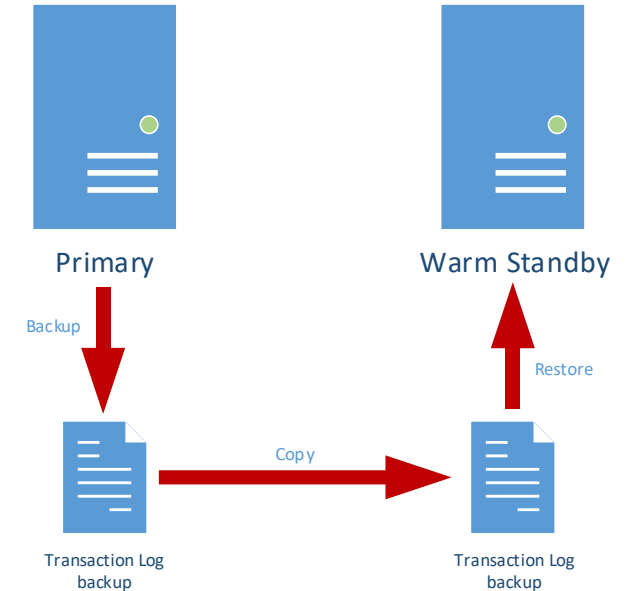
- Same HADR options exists in both Linux and Windows.
 - Except no database mirroring in Linux (deprecated on windows).



Failover Cluster Instance



Availability Groups



Log Shipping

SQL Server On Linux

Failover Clusters

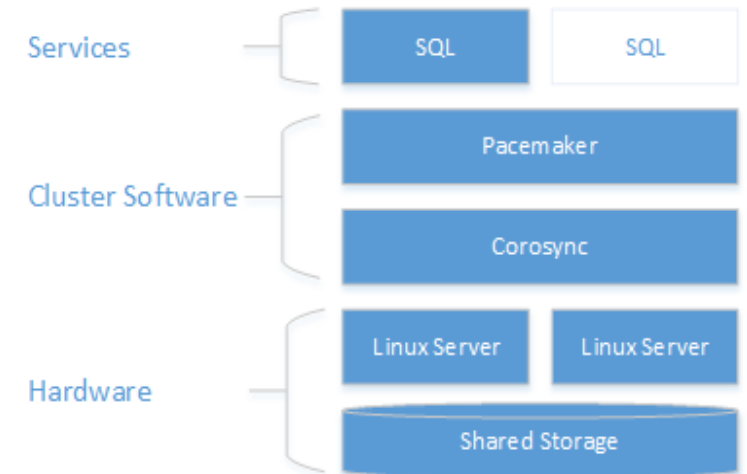
Failover Clusters (FCI) work basically the same in both Linux and Windows.

Key differences is in the platform clustering technology.

Pacemaker is the supported clustering technology in Linux.

No separate network name/IP resource for SQL.

- Connectivity leverages the name and IP specified for the cluster
- Must be registered in DNS



SQL Server On Linux

Availability Groups

Failover must be done via the cluster in Linux whereas Windows AGs are failed over via SQL Server.

Listener IP and name must be manually registered in DNS.

Cluster type of External.

Currently no support for DTC on Linux based AGs.

Three replicas is the required minimum in Linux due to the need for quorum.

SQL Server On Linux

Tools

Command line:

- Sqlcmd
- Bcp
- Mssql-conf

SSMS and SSDT support SQL on Linux remotely from a Windows client

Azure Data Studio

SQL Server On Linux

Enhancements since SQL 2019

Replication Support

Change Data Capture (CDC) Support

Distributed Transaction Coordinator (MSDTC) Support

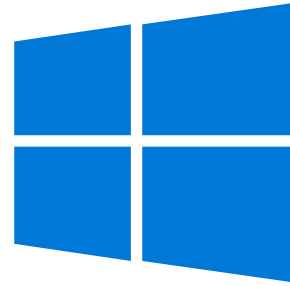
Open LDAP Support for Third-party AD Providers

Machine Learning Support

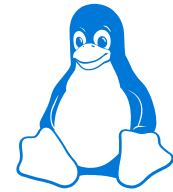
Tempdb files auto-config

PolyBase on Linux

Windows



Linux



Containers and Clusters



OPENSIFT



kubernetes

Questions?



Knowledge Check

What are some of the Linux platforms that are supported to install SQL Server?

Can you configure Availability Groups in SQL Server on Linux 2022?

Which SQL Server services are not supported in SQL Server on Linux?

Which version of SQL Server On Linux introduced support for Replication?

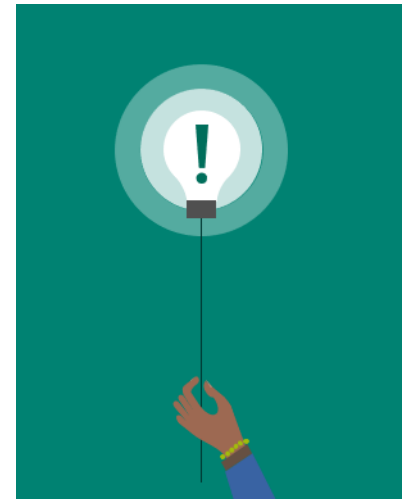
Name some of the tools that can be used to work with SQL Server on Linux

Lesson 3: SQL Server on Containers

Objectives

After completing this learning, you will be able to:

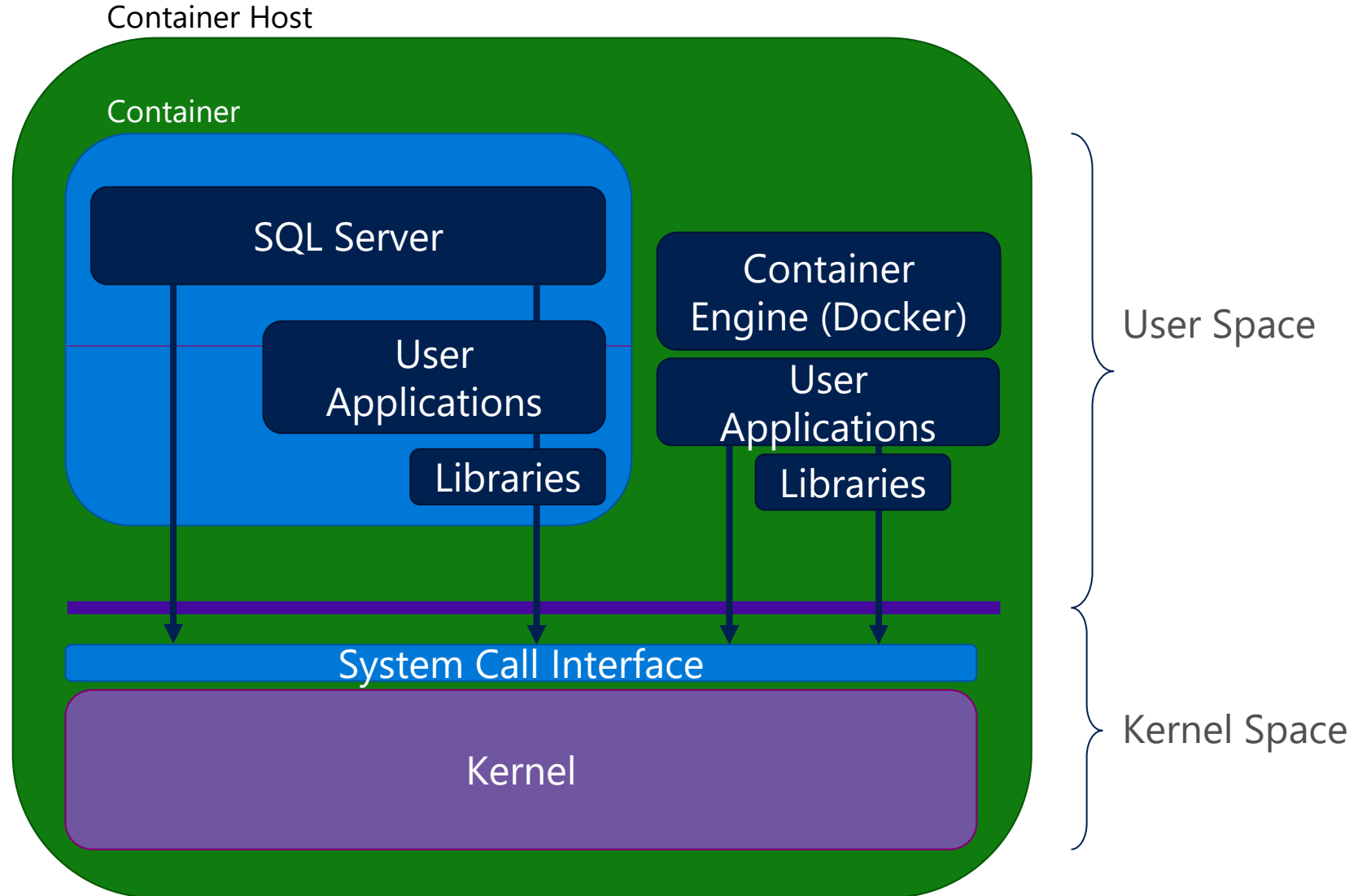
- Understand the use cases for containers and dockers.



What are Containers?

Lightweight, standalone, executable package of software that includes everything needed to run an application: code, runtime, system tools, system libraries and settings

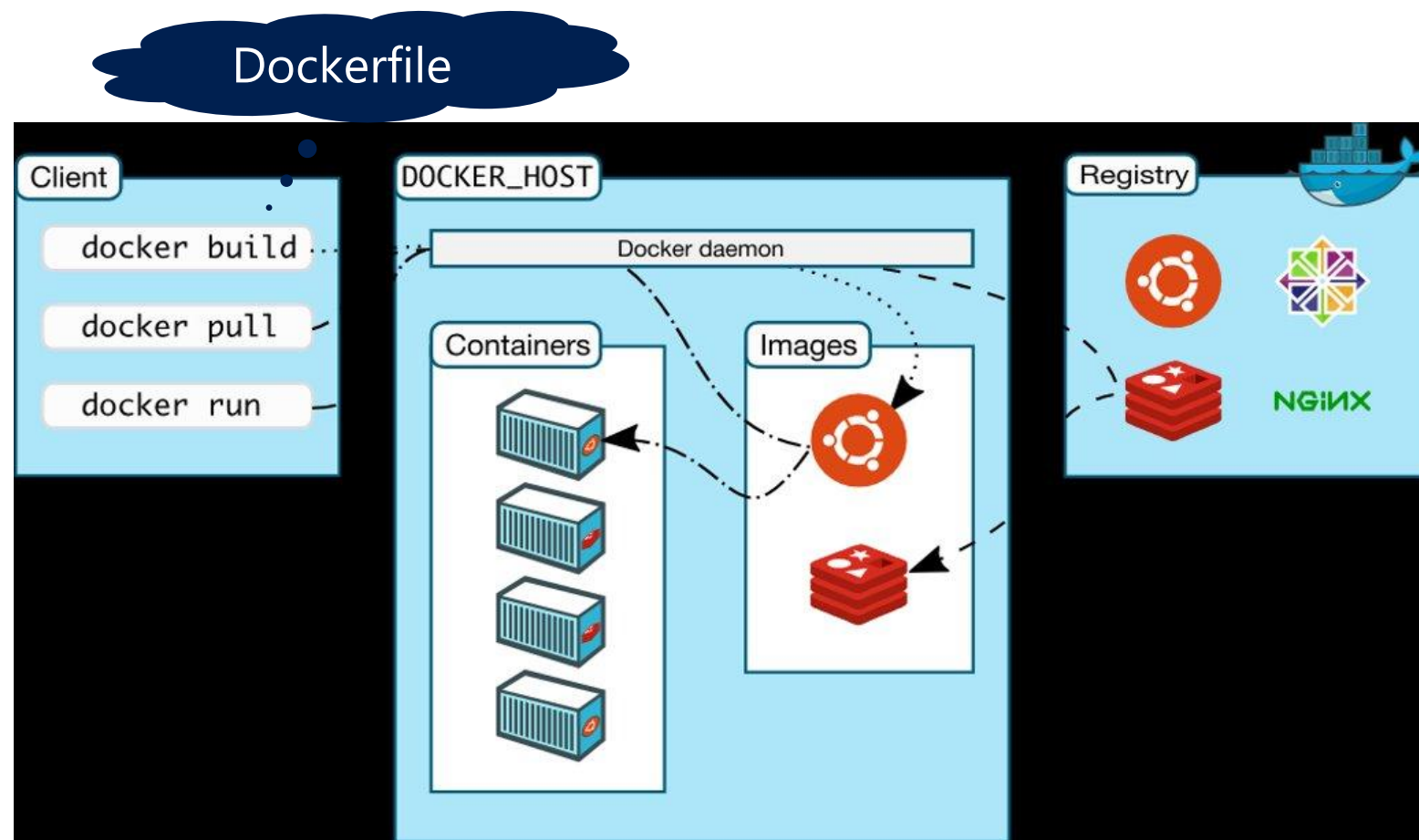
Isolated user space on top of operating systems kernel



What is Docker?

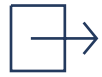
Most popular open Source container engine used for automating the deployment of applications as portable, self-sufficient containers.

Uses copy-on-write union file system for its backend storage.



Why Use Containers?

Portable



Move container images between Test, Dev and Prod environments (DevOps)



Lightweight

Reduced disk, CPU, and memory footprint



Consistent

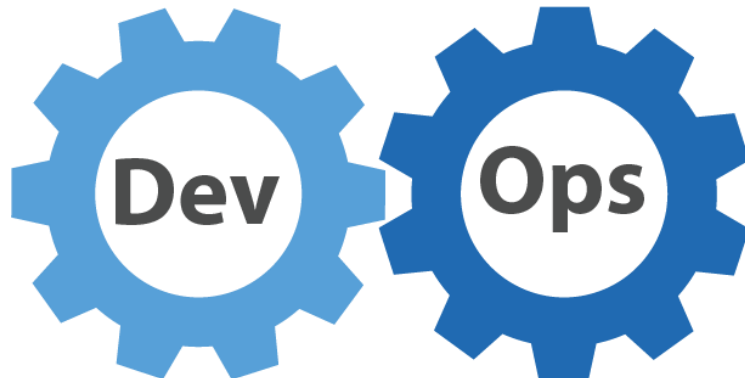
Consistent image of SQL Server, scripts, and tools



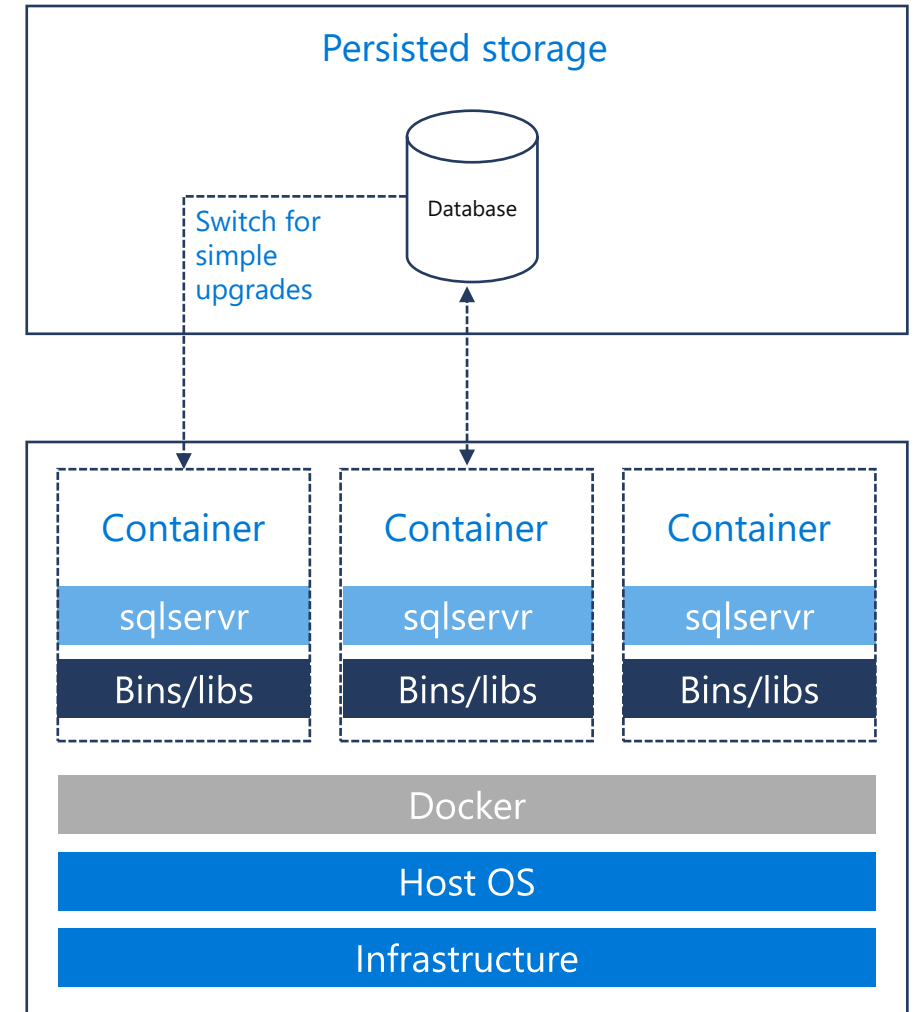
Efficient

Faster deployment, reduced patching, and less downtime

No Installation Required



Container configuration



Container Images

VM = emulate hardware and load entire OS

Containers = isolation but cooperate with host kernel

Image

Layered snapshot of a set of files structured in a filesystem

Built with a **Dockerfile** using docker build or APIs

Most built with a **base image** of an OS

The base image OS must be the **same kernel as host**

Container

Runtime instance of an image

Run as many you like based on the same image

Isolated but can communicate with the "outside"

Has **access** by default to all CPU and memory resources

Each container has **writable layer** but **share read-only image layer**

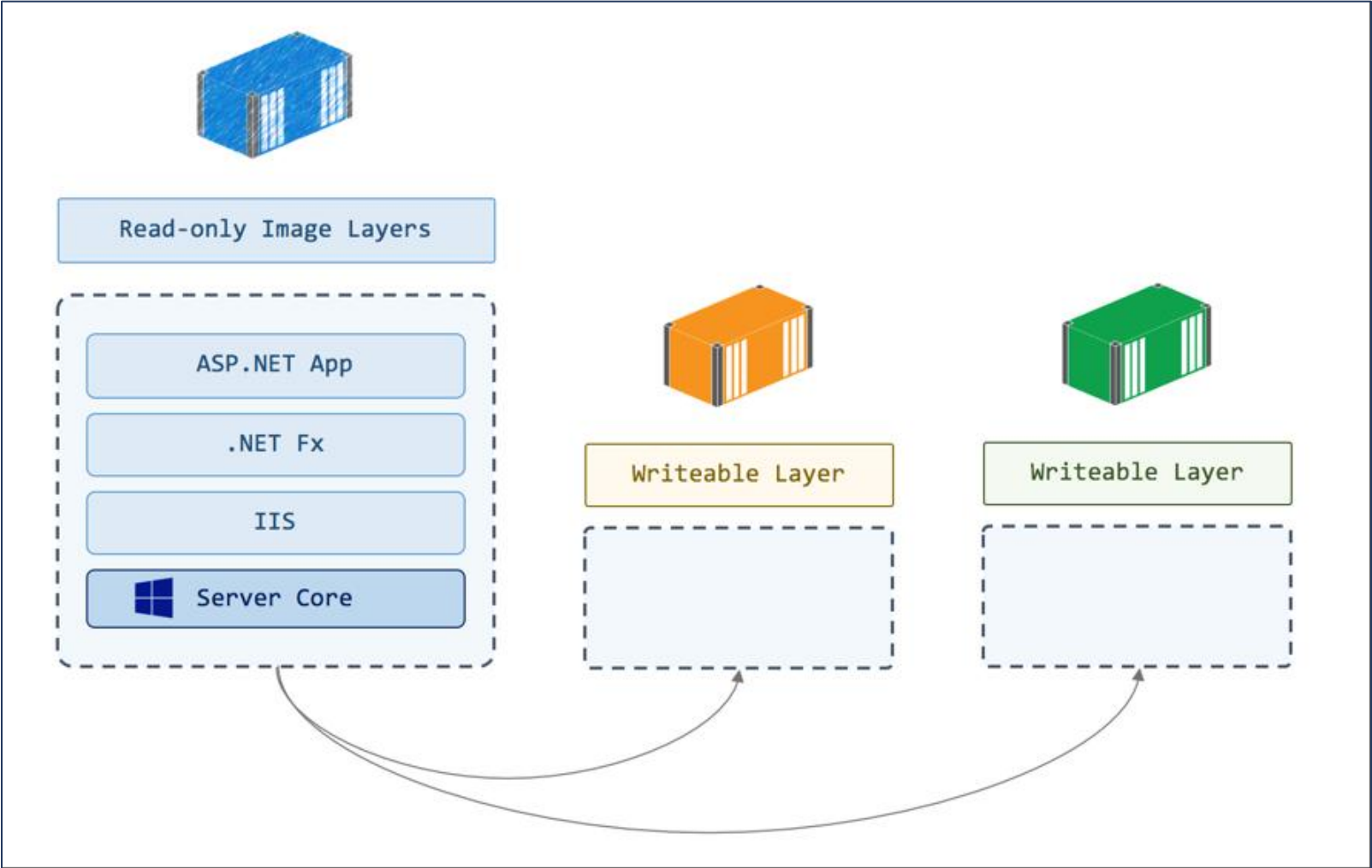
Start and stop do not affect writable layer

Remove deletes writable layer

Volumes provide persistence on host

Containers

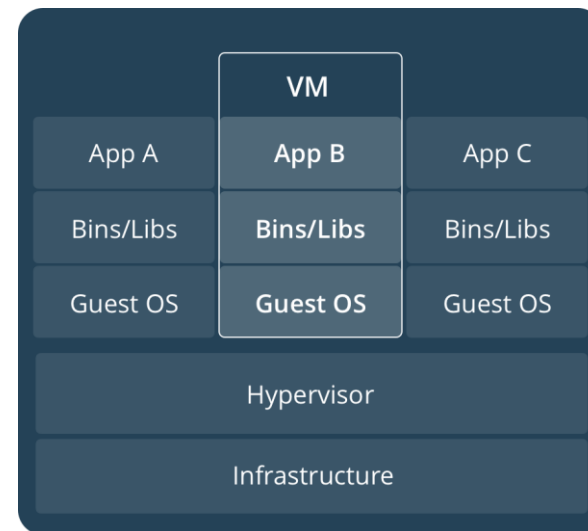
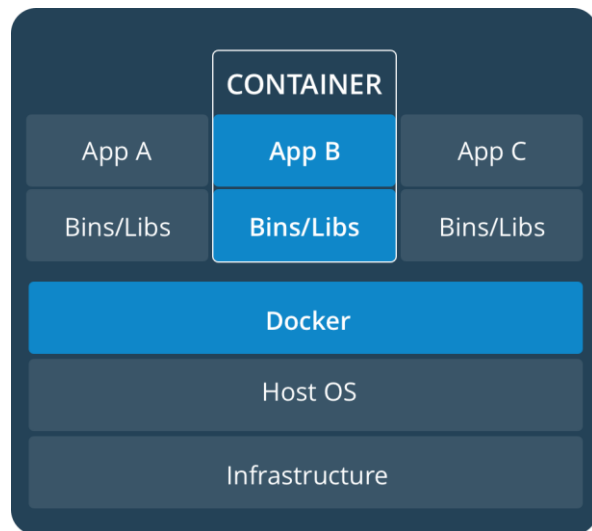
Layers and Copy on Write



Virtual Machines Versus Containers

A container runs natively on Linux and shares the kernel of the host machine with other containers. It runs a discrete process, taking no more memory than any other executable, making it lightweight.

By contrast, a virtual machine (VM) runs a full-blown “guest” operating system with virtual access to host resources through a hypervisor. In general, VMs provide an environment with more resources than most applications need.



SQL Server Containers

Short lived SQL server instances for Dev and QA

Reduces the number of SQL Server host or virtual machines

- Significant savings in VM maintenance

A mounted 1 TB database is delivered in seconds and can be deleted and replaced as needed

Updates to the data or code can be incorporated into a new container image, and environments can be refreshed in minutes

Each container includes a private file system, and users are limited to their container and file system

You can create your own docker images with custom apps, tools and configurations embedded along with SQL Server

SQL Server Containers

Use Case Scenarios

Continuous Integration/Continuous Deployment (CI/CD) pipelines

Dev/Test - Parallel testing made fast and easy

Time bound applications

Multi-OS development, test, and production environments

Running a SQL Container

sudo docker run

-e 'ACCEPT_EULA=Y' -e
'MSSQL_SA_PASSWORD=Sql2022isfast'

-p 1401:1433

-v sqlvolume:/var/opt/mssql

--hostname sql1

--name sql1

-d

mcr.microsoft.com/mssql/server:2022-latest

[environment variables](#)
(Ex. enable SQLAgent)

Your app connects to
port 1401

Map default directory
to host volume for
persisted storage

@SERVERNAME

Use with docker client

Run process in background.
Remove this to debug

SQL Server 2022
latest image

Questions?



Knowledge Check

Can I use SQL Server Management Studio (SSMS) or Azure Data Studio to connect to SQL Server in a container and run queries?

How can you create multiple instances of SQL Server on Linux OS?

What allows multiple containers to share access to the same underlying image and yet have their own data state?

Lesson 4: SQL Server in Microsoft Azure

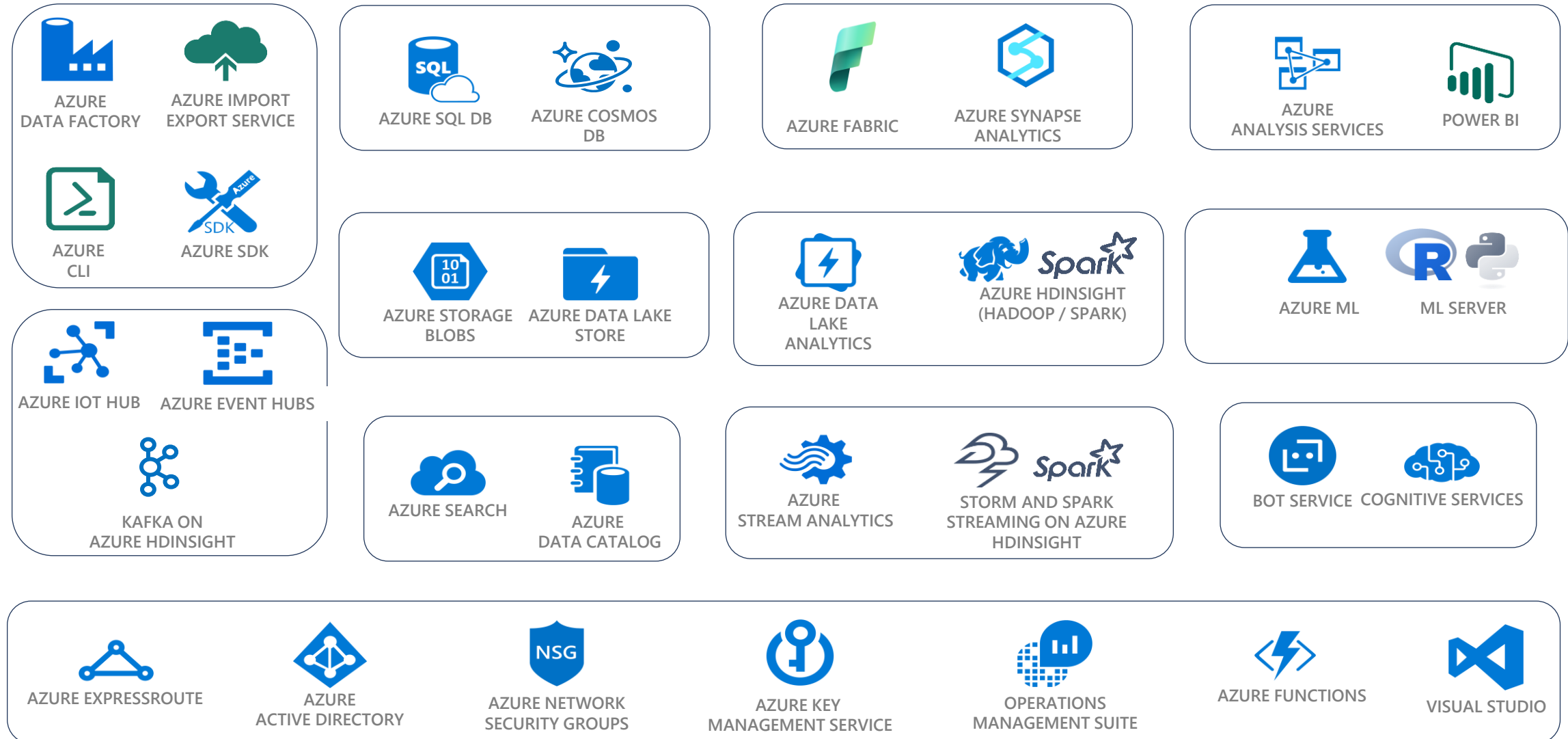
Objectives

After completing this learning, you will be able to:

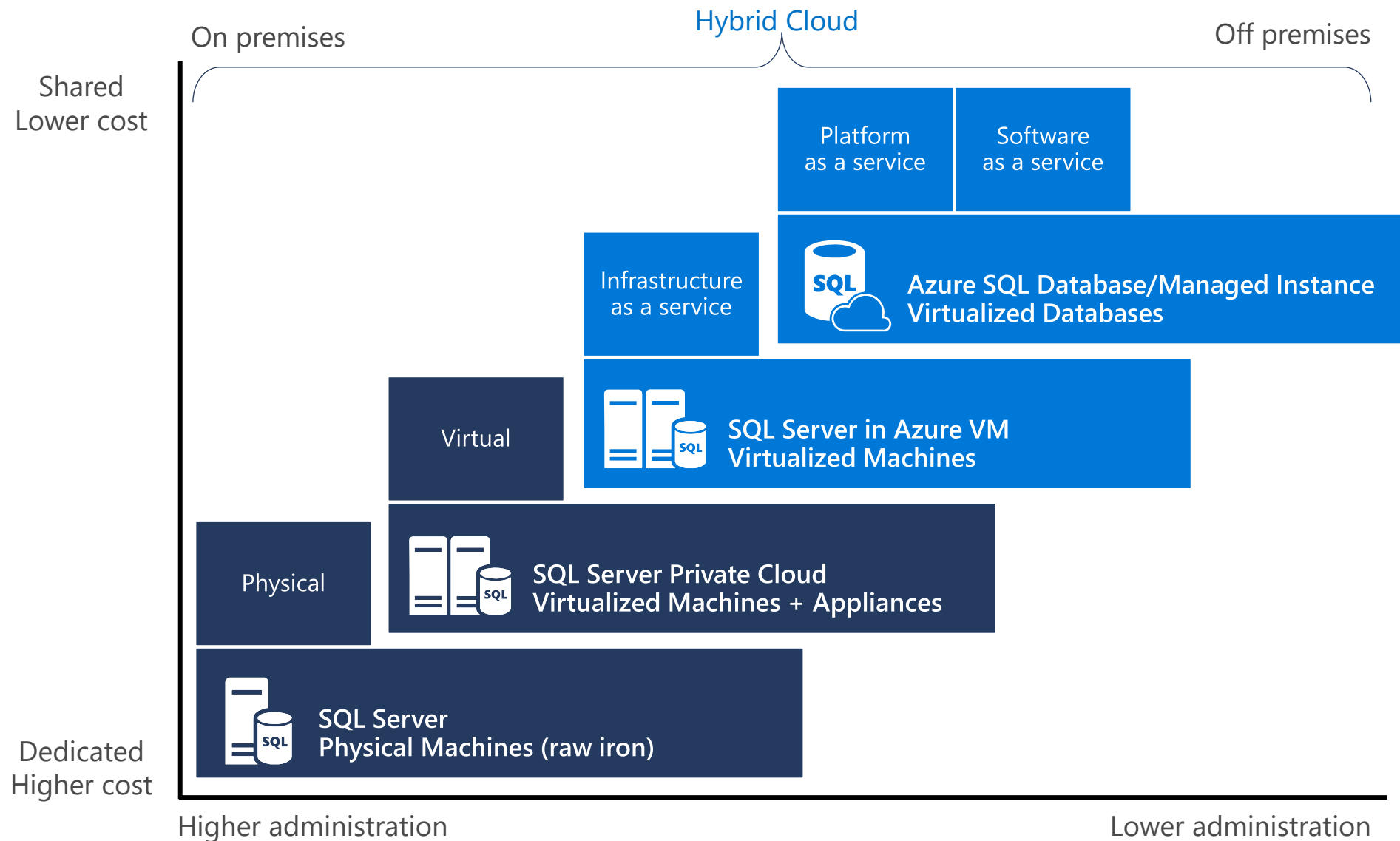
- Understand SQL Server services in Microsoft Azure.
- Compare and Contrast SQL Services in Azure to figure out the right service to choose for your workloads.



Microsoft Azure Data Platform Landscape



SQL Server Data Platform Evolution



SQL Server Services in Microsoft Azure

Platform as a Service (PaaS): Database

- All hardware, software, and infrastructure are taken care of by Azure.
- Azure provides HADR, monitoring, and intelligence.

Platform as a Service (PaaS): Managed Instance

- Fully fledged SQL instance with nearly 100% compatibility with on-premises.
- Azure manages the hardware as well as the OS.

Infrastructure as a Service (IaaS): Virtual Machine

- Hardware and Infrastructure are taken care of by Azure.
- You own SQL Server and OS as well as monitoring and HADR.
- Many sizing options.
- Pre-created library of virtual machines.

Azure SQL Database

Deployment Options



Azure SQL Database

Managed instances

Best for most lift-and-shift migrations to the cloud



Single instance

- SQL Server surface area (vast majority).
- Native virtual network support.
- Fully managed service.



Instance pool

- Resource sharing between multiple instances to price optimize.
- Simplified performance management for multiple databases.
- Fully managed service.

Databases

Best for modern cloud applications. Hyperscale and serverless options are available



Single database

- Hyperscale storage (up to 100TB).
- Serverless compute.
- Fully managed service.



Elastic pool

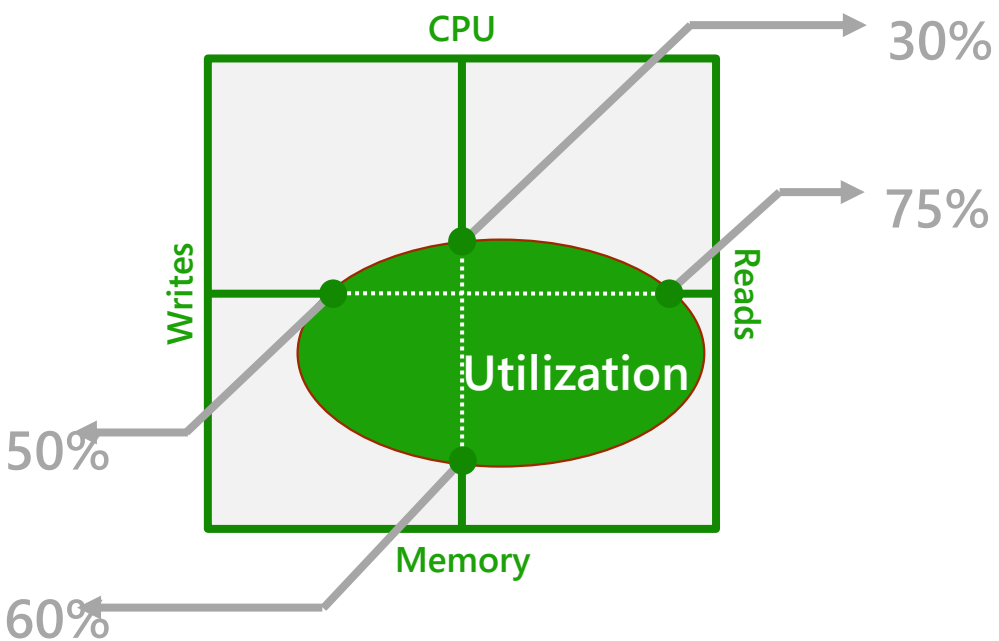
- Resource sharing between multiple databases to price optimize.
- Simplified performance management for multiple databases.
- Fully managed service.

Azure SQL Database

Database Transaction Unit (DTU)

- Represents the relative power (resources) assigned to the database
- Blended measure of CPU, memory, and read-write rates
- Simplifies talking about database performance
- DTU versus eDTU

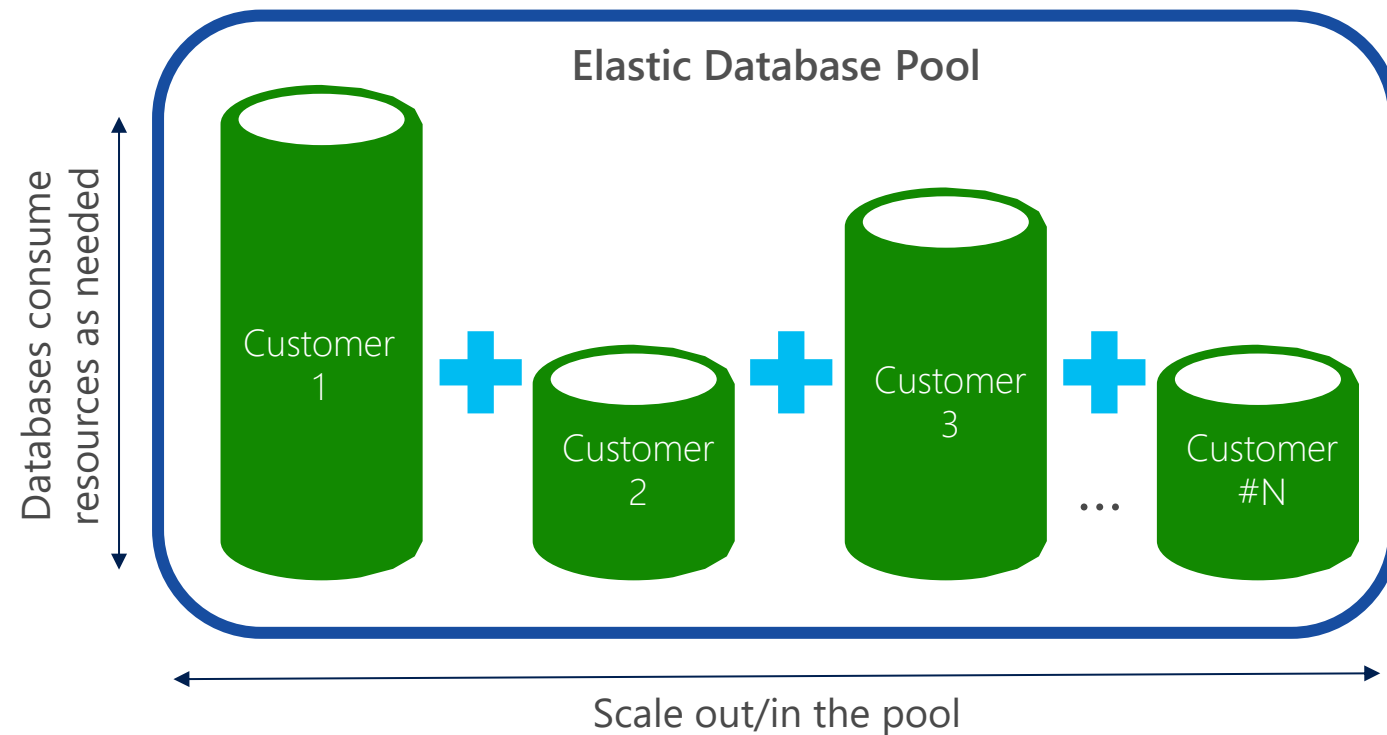
Purchasing model/ Deployment model	DTU based	vCore based
Single Database	✓	✓
Elastic Pool	✓	✓
Managed Instance	✗	✓
Hyperscale service tier (single databases)	✗	✓



Azure SQL Database

Elastic Pools

- Ensure that databases get the performance resources they need, when they need it, while providing a simple resource allocation mechanism within a predictable budget.



Azure SQL Database

Managed Instance (MI)

PaaS Benefits

- No management overhead for managing underlying infrastructure
- “Relatively quick” provisioning and service scaling
- Automated patching and version upgrade

Management

- Azure Resource Manager (ARM) API for automating service provisioning and scaling
- Azure portal functionality for manual service provisioning and scaling

Business Continuity

- Built in high availability
- 99.99% uptime SLA
- Data protected with automated backups

Security and Compliance

- Isolated environment (VNet integration, single-tenant service, dedicated compute and storage)
- Azure AD authentication, single sign-on support
- SQL auditing and threat detection

Azure SQL Database – Managed Instance

Key Differences between SQL Server and Managed Instances

High-availability is built in and pre-configured in MI

Automated backups

MI does not allow access to the file system

MI supports Azure EntraID (ex. Azure AD) authentication

MI automatically manages filegroup (called XTP) for databases containing In-Memory OLTP objects

SSIS packages hosted in MI SSIS catalog (SSISDB) are executed on a managed Azure-SSIS Integration Runtime (IR) in Azure Data Factory (ADF)

Azure SQL Database

Key Differences between MI and Azure SQL Database

MI does not support:

- Automatic index tuning
- Geo-replication
- SQL Data Sync
- Event notifications
- Elastic Pools

Azure SQL DB does not support:

- User-initiated backups
- Change Data Capture (below S3 tier and Basic)
- Common language runtime (CLR)
- Cross-database queries and transactions
- Database mail
- Distributed partition views
- Event notifications (Use Alerts instead)
- Multiple filegroups
- Linked servers
- OPENDATASOURCE, OPENQUERY
- Resource Governor
- Service Broker
- SQL Server Agent (use Elastic Jobs instead)
- SQL Server Profiler (use Extended Events)
- Most system SPs

Azure SQL Database

Migration Methods

Database Migration Assistant

Azure Migrate

Azure SQL Migration extension for Azure Data Studio

Transactional Replication

Import / Export Service / BACPAC

Backup and Restore

Bulk Load

Azure Data Factory

SQL Data Sync

Demonstration

Create an Azure SQL Database
and connect from SSMS or
Azure Data Studio



Azure Virtual Machines

Flexibility of virtualization without having to buy and maintain the physical hardware

Customers responsible for maintaining the virtual machine such as configuring and patching

Three ways to provision SQL Server on VMs:

Certified Images on Azure Market Place



Pay-as-you-go
images



Bring your own
license images



Select Azure VM and
self-install SQL Server



Maintained and refreshed by Microsoft



SQL IaaS Extension for automated configuration and administration

Azure Virtual Machines

Storage



Operating System Disk. Registered as a SATA drive and labeled as the C: drive by default. This disk has a maximum capacity of 4,095 gigabytes (GB). If partitioned with MBR only 2 TB possible to use.



Temporary disk. It is automatically created and it is labeled as the D: drive by default and it is used for storing pagefile.sys. Its size varies, based on the size of the virtual machine.



Data disks. Registered as SCSI drives and labeled with a letter that you choose. Disk sizes vary by chosen disk type (32TB – 64TB). The size of the virtual machine determines how many data disks you can attach to it and the type of storage you can use to host the disks.

Azure Virtual Machines

Disks

A variety of disks possible, SSD (Ultra, Premium V2, Premium, Standard) and HDD (Standard)

Azure Premium Storage supports Virtual Machine disks that can be attached to various Azure VMs on diverse D, E, LS Series

Premium Storage Account supports Locally Redundant Storage (LRS) and Zone Redundant Storage (ZRS)

Limits:

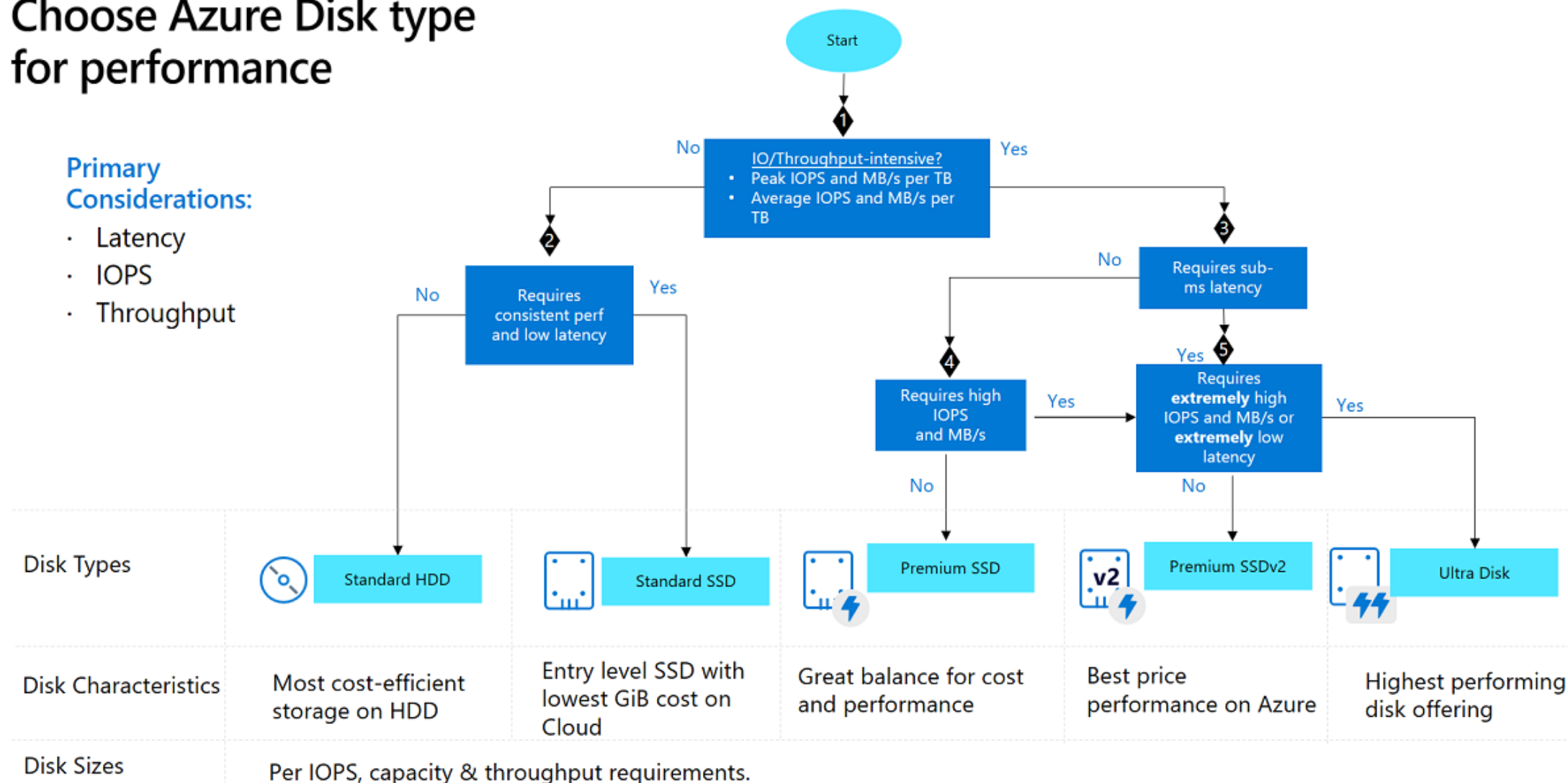
- HDD 32 TB per Disk, SSD up to 64 TB per Disk
- Throughput from 500MB/sec (HDD) up to 4000 MB/sec (Ultra SSD)
- MaxIOPS from 2000 (HDD) up to 160000 with Ultra SSD

Azure VM Disk Choice

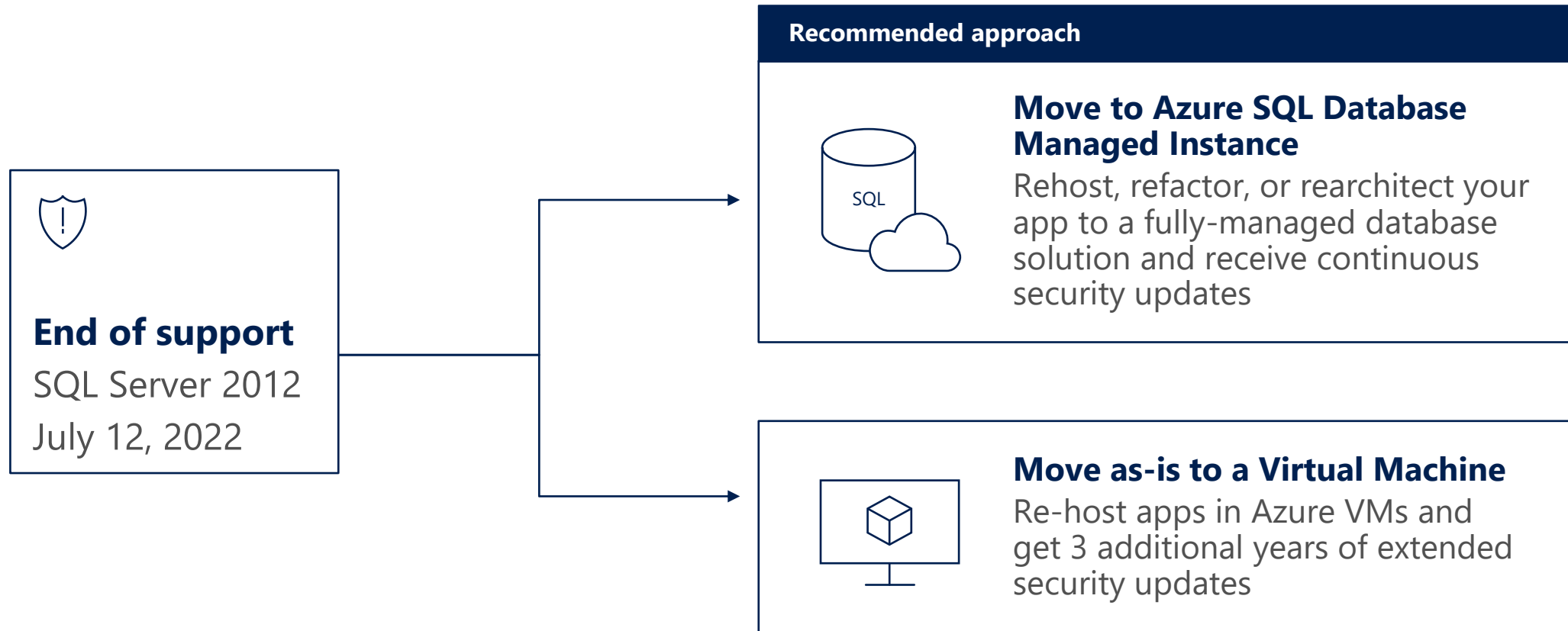
Choose Azure Disk type for performance

Primary Considerations:

- Latency
- IOPS
- Throughput



Migration options for SQL Server 2012 end of support



SQL 2008 and SQL 2008R2 are no longer available on Azure Marketplace

Questions?



Knowledge Check

Which options exists for running a SQL Server database in Microsoft Azure?

Is it important to test your solution performance when implementing databases on Microsoft Azure? Why?

What is the difference between DTU and eDTUs?

What are end of support options for SQL Server 2012?

