Tips for Getting Started with the Azure Data Platform

Practical Tips and Lessons Learned

April 17, 2018

Melissa Coates

Solution Architect, BlueGranite

Microsoft Data Platform MVP

Blog: www.sqlchick.com

Twitter: @sqlchick

Agenda

Goals for this session:

1. Key concepts for using Azure as a data platform
2. Share suggestions and lessons learned through a series of tips and demos

Lots of important things are out of scope though!

- Database migration techniques
- Security
- Azure Stack & private cloud deployments
- Networking & connectivity
- Troubleshooting, performance tuning & growth
- Installation and configuration details
- Monitoring & alerting

Azure services & features change very frequently, so be sure to verify details

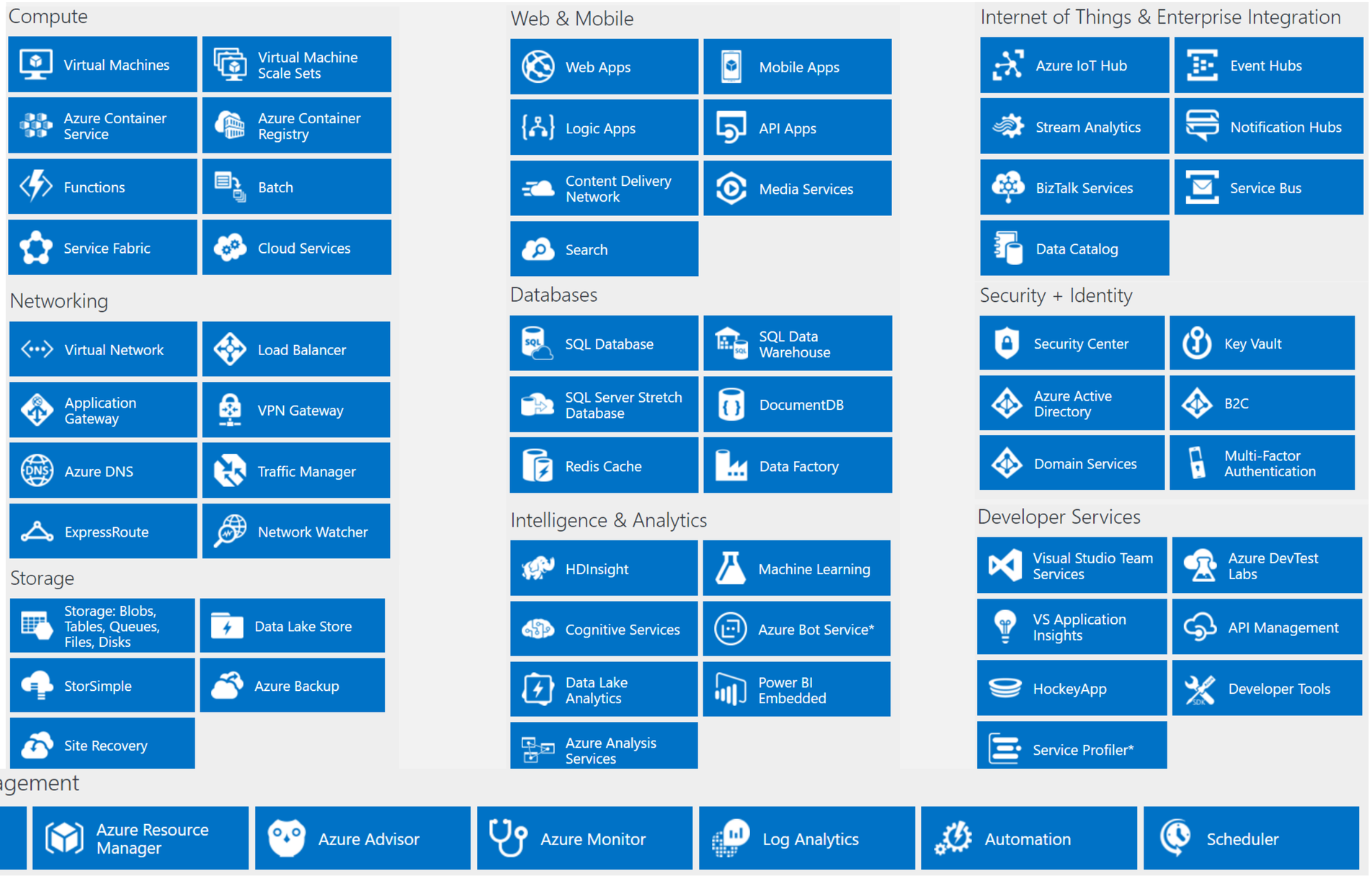
Tip #1

Know the What & Why for Moving to the Cloud

Azure

A public cloud computing platform and infrastructure for building, deploying, and managing MSFT-specific and third party software and services through a global network of Microsoft-managed datacenters

Source: http://azureplatform.azurewebsites.net/



Why the Cloud

Common Pros

- ✓ Eliminate/reduce data center management
- ✓ Elasticity (scaling up/down)
- ✓ Self-service provisioning of services
- ✓ Ease of experimentation (agility)
- ✓ Faster time-to-market
- ✓ Easier high availability and disaster recovery
- ✓ Subscription-based operating expenses (rather than capital expenses with large up-front investment)

Your goals for the cloud affect the tradeoffs & decisions you will make on control, complexity, cost, performance & security

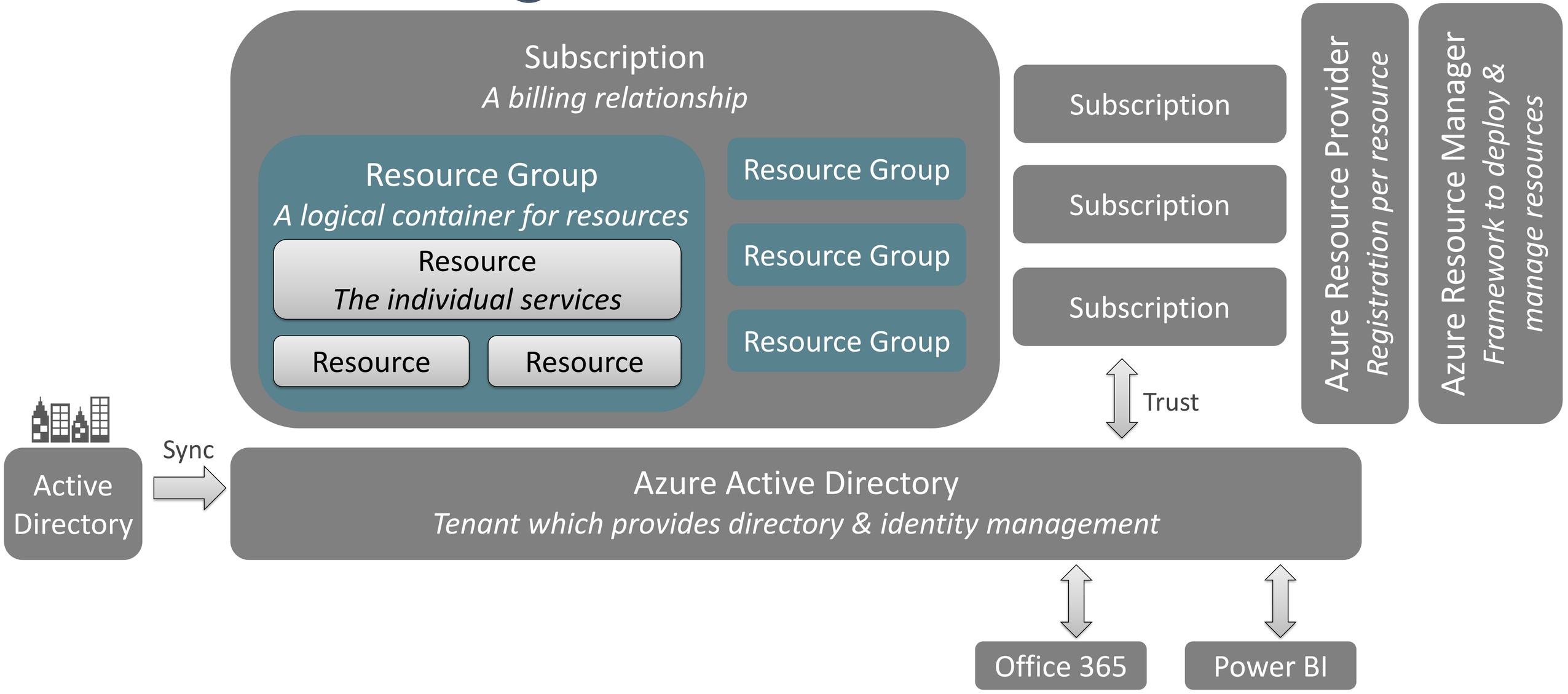
Common Concerns

- ✓ Uptime guarantees
- ✓ Performance
- ✓ Security
- ✓ Compliance, regulations, legal
- ✓ Sharing of resources (multi-tenancy; noisy neighbors)
- ✓ Data and intellectual property privacy
- ✓ Vendor lock-in/dependency
- ✓ Connecting legacy systems (hybrid/on-prem)
- ✓ Sprawl of self-provisioned services
- ✓ Lack of cloud expertise
- ✓ Complexity
- ✓ Cost
- ✓ Difficult to estimate cost up-front

*The cloud is not *always* easier*

*The cloud is not *always* cheaper*

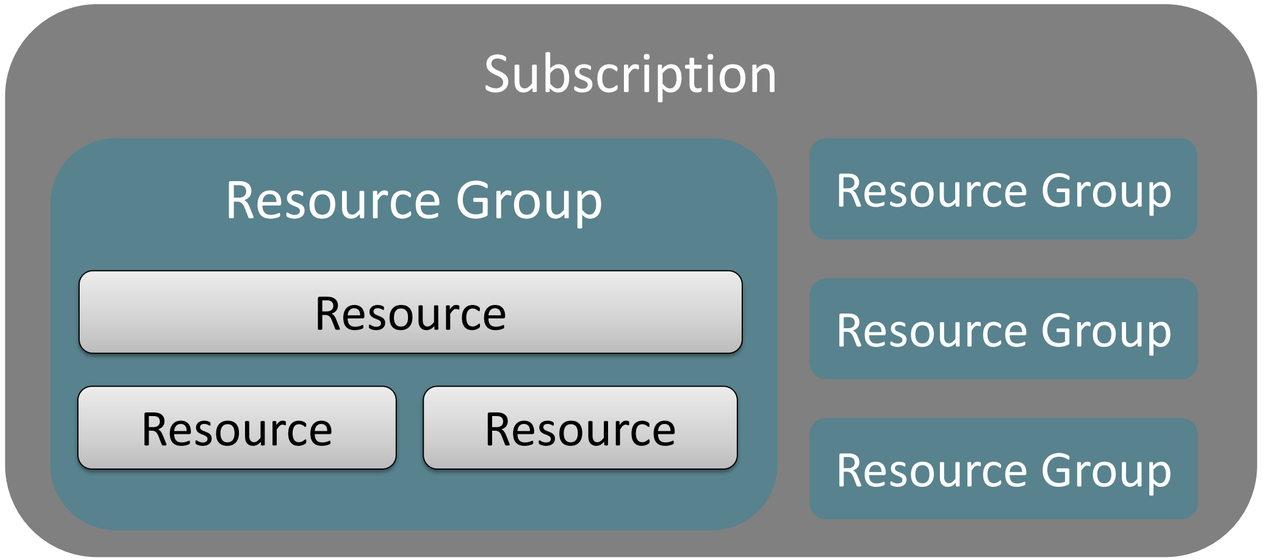
The Azure Lingo



Tip #2

Plan for Resource Groups Very Carefully

Resource Groups - Considerations



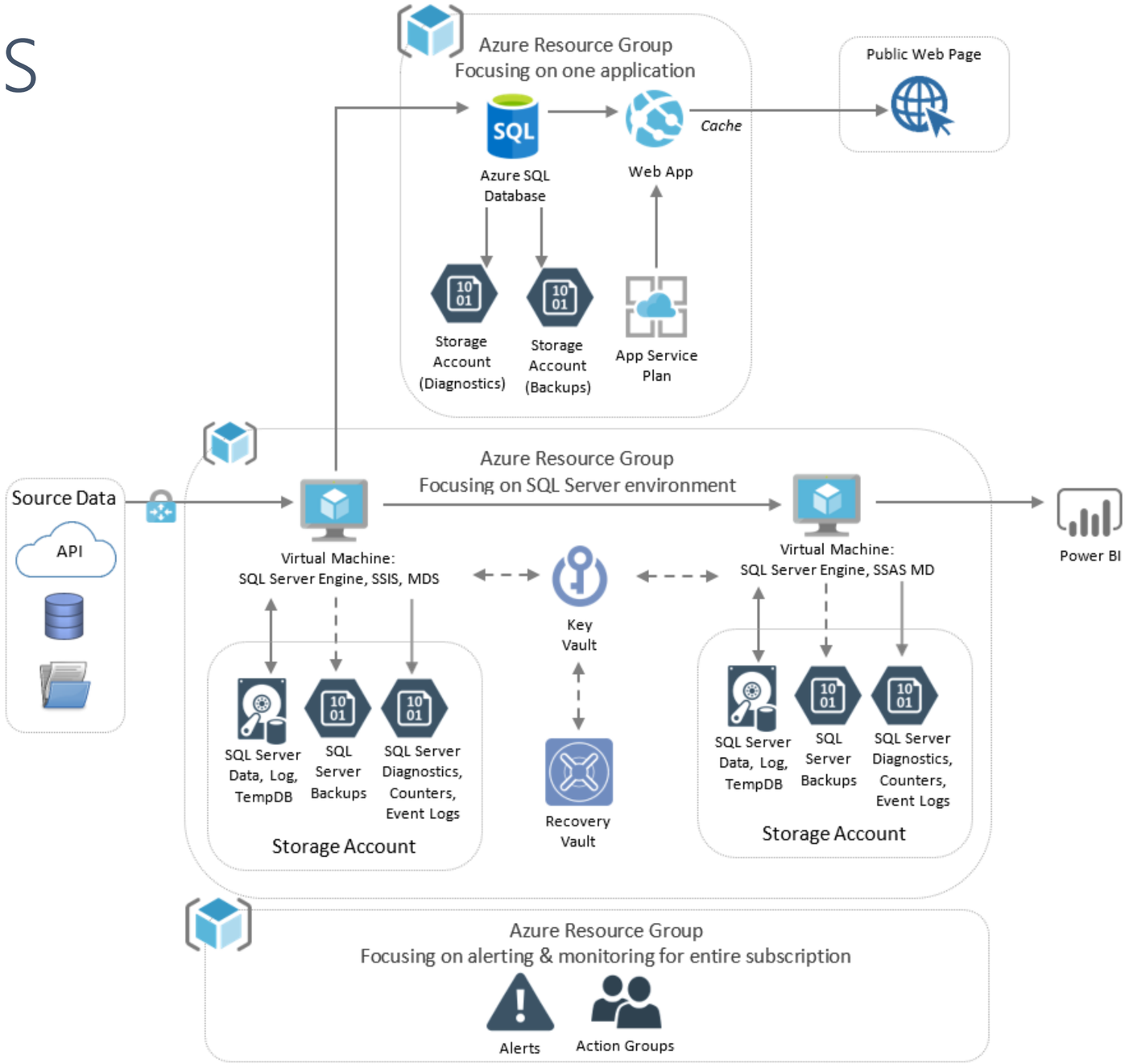
Considerations:

- ✓ Logical organization by purpose
- ✓ Permissions/security boundaries
- ✓ Policies & management of resources as a group
- ✓ Deployment lifecycle -- RGs are the default scope for declarative ARM deployment scripts
- ✓ Location of services
  - Geographic location of data
  - Proximity to business users
  - Co-location of related resources
  - Minimizing latency
  - Minimizing data egress charges
  - Not all resources/services available in every region
  - Backup/recovery considerations

Resource Groups

Suggestions:

- ✓ Plan very carefully for resource groups
- ✓ Keep resource groups more narrow than broad
- ✓ Select the region (location) carefully



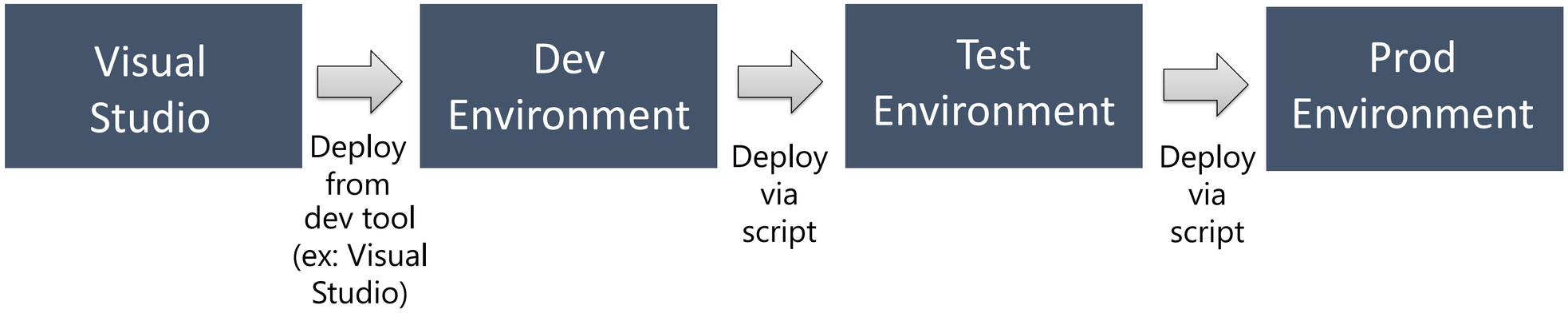
Tip #3

Plan Deliberately for Dev, Test, Prod Environments

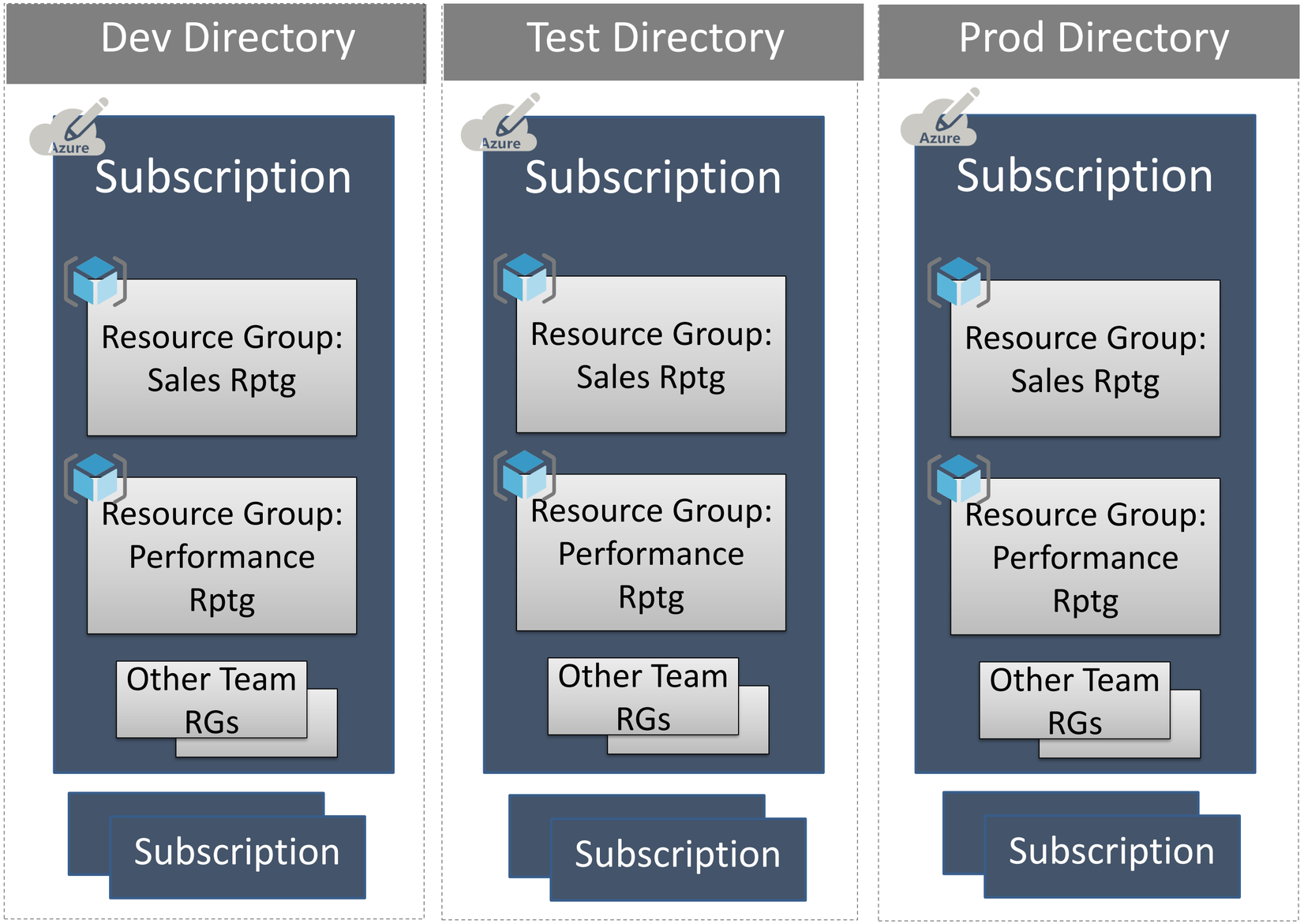
Separation of Dev, Test, Prod Environments

Most commonly environments are segregated by:

1. Directory, or
2. Subscription, or
3. Resource Groups,
4. A combination of 1 and 3, or 2 and 3



Option: Separate By Directory



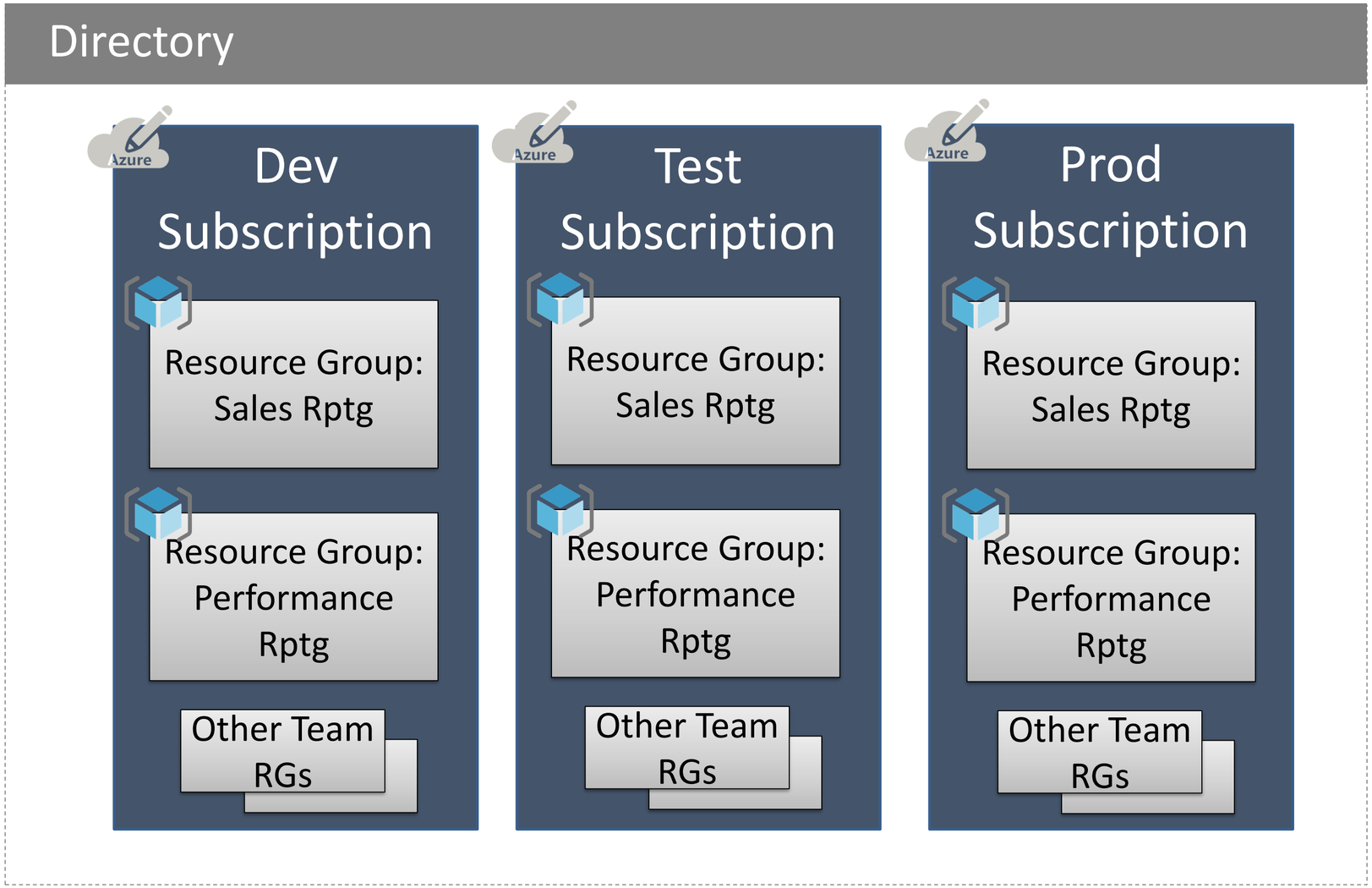
Pros:

- ✓ Clear boundary
- ✓ Offers the most scalability

Cons:

- ✓ More infrastructure to manage
- ✓ A lot of objects intermixed in a subscription - need clear resource group names and/or tags to tell what belongs to which team
- ✓ More complex AAD configuration
- ✓ Separate virtual networks & VPN set up across subscriptions

Option: Separate By Subscription



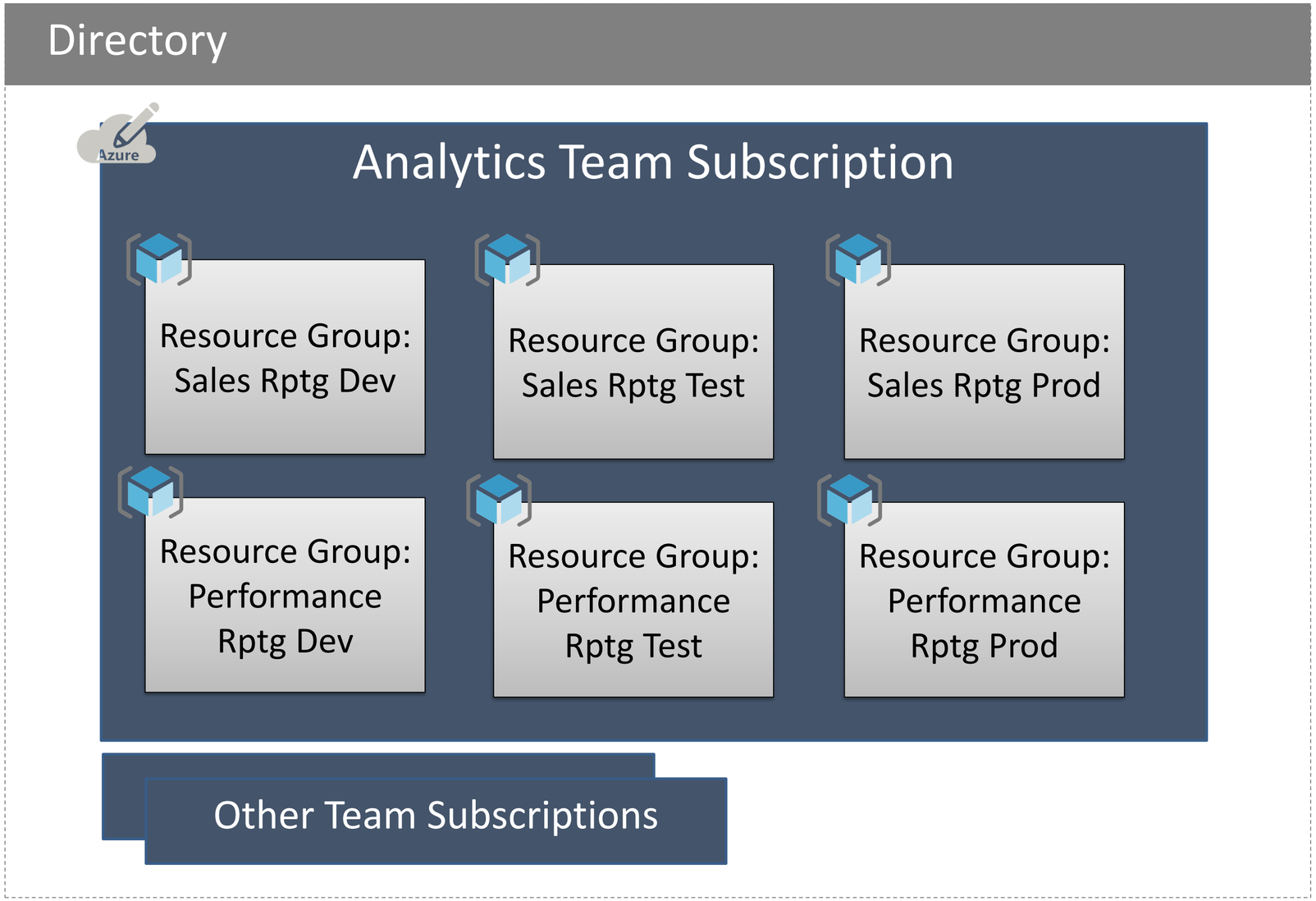
Pros:

- ✓ Clear boundary

Cons:

- ✓ Cannot always provision new resources if you're just an owner at the resource group level
- ✓ A lot of objects intermixed in a subscription - need clear resource group names and/or tags to tell what belongs to which team
- ✓ Separate virtual networks & VPN set up across subscriptions

Option: Separate by Resource Group



Pros:

- ✓ Isolation of subscriptions by the team who owns/manages
- ✓ Co-admin privileges easier to delegate at the subscription level for each team
- ✓ Billing segregation
- ✓ More subscriptions: might be less likely to hit Azure subscription limits

Cons:

- ✓ More risk because Prod is mixed in with Dev & Test (mitigate w/ clear names & diff IDs)
- ✓ Separate virtual networks & VPN set up across subscriptions

Demo

Resource Groups, Resources, and Resource Providers

Tip #4

Use Clear Naming Conventions

Naming Conventions

Purpose → Type of Service → Environment

- ✓ Type of service in the name helps with logging/metrics in monitoring scenarios
- ✓ Environment as the suffix makes any concatenations easier within scripts
- ✓ Prod is enumerated because we separate Dev/Test/Prod within one subscription
- ✓ No dashes since all services don't allow them
- ✓ Camel case if the service allows it; otherwise lower case

Resource Groups

InternalReportingRGDev
InternalReportingRGTest
InternalReportingRGProd

Virtual Machines

BISQLVM1Dev
BISQLVM1Test
BISQLVM1Prod

Storage Accounts

BISQLVM1DataStdStrgDev
BISQLVM1BckStdStrgDev
BISQLVM1DiagStdStrgDev

More info on naming conventions: http://www.sqlchick.com/entries/2017/6/24/naming-conventions-in-azure

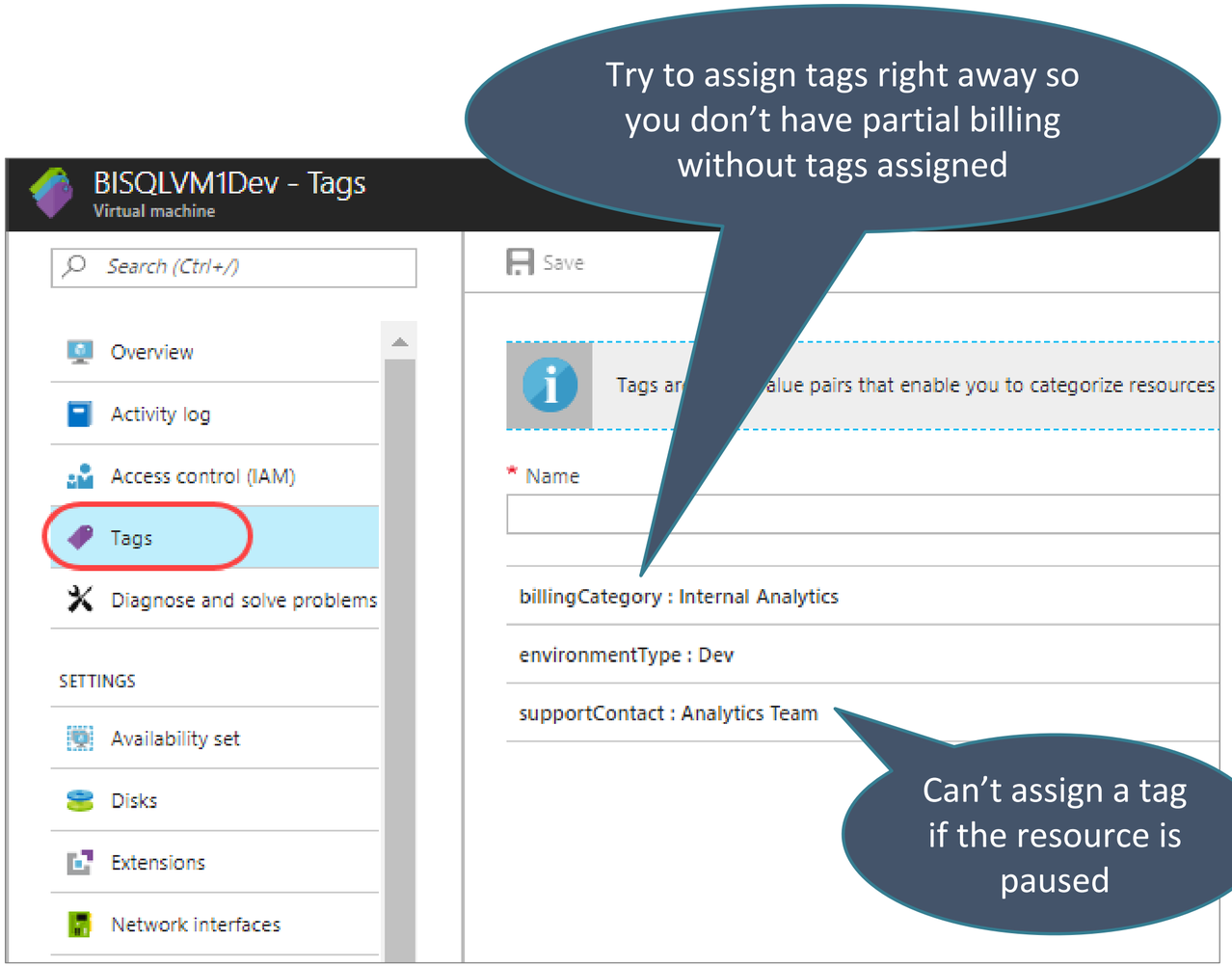
Tip #5

Use Tags to Categorize Billing Charges & Other Metadata

Standard Tags

Tags are very helpful for:

- ✓ Billing or cost center categories
- ✓ Environment names
- ✓ Project or system
- ✓ Purpose or application
- ✓ Team, department, business unit
- ✓ Who owns or supports a resource
- ✓ Release or version #s (ex: testing infrastructure)
- ✓ Archival date (ex: if needed only temporarily)
- ✓ Patching or maintenance window or SLA
- ✓ Which customer it applies to (ex: if an ISV)
- ✓ etc…



Meter Name	Consumed Service	Resource Group	Tags
"Standard IO - Block Blob Read Operation Units (in 10,000s)"	"Microsoft.Storage"	"internalreportingrgdev"	"{"billingCategory":"Internal Analytics","environmentType":"Dev","supportContact":"Analytics Team"}"
"Standard IO - Table Write Operation Units (in 10,000s)"	"Microsoft.Storage"	"internalreportingrgdev"	"{"billingCategory":"Internal Analytics","environmentType":"Dev","supportContact":"Analytics Team"}"
"Standard IO - Page Blob/Disk (GB)"	"Microsoft.Storage"	"internalreportingrgdev"	"{"billingCategory":"Internal Analytics","environmentType":"Dev","supportContact":"Analytics Team"}"
"Standard IO - Table (GB)"	"Microsoft.Storage"	"internalreportingrgdev"	"{"billingCategory":"Internal Analytics","environmentType":"Dev","supportContact":"Analytics Team"}"

More info on tags: http://www.sqlchick.com/entries/2017/7/15/powershell-for-assigning-querying-tags-in-azure

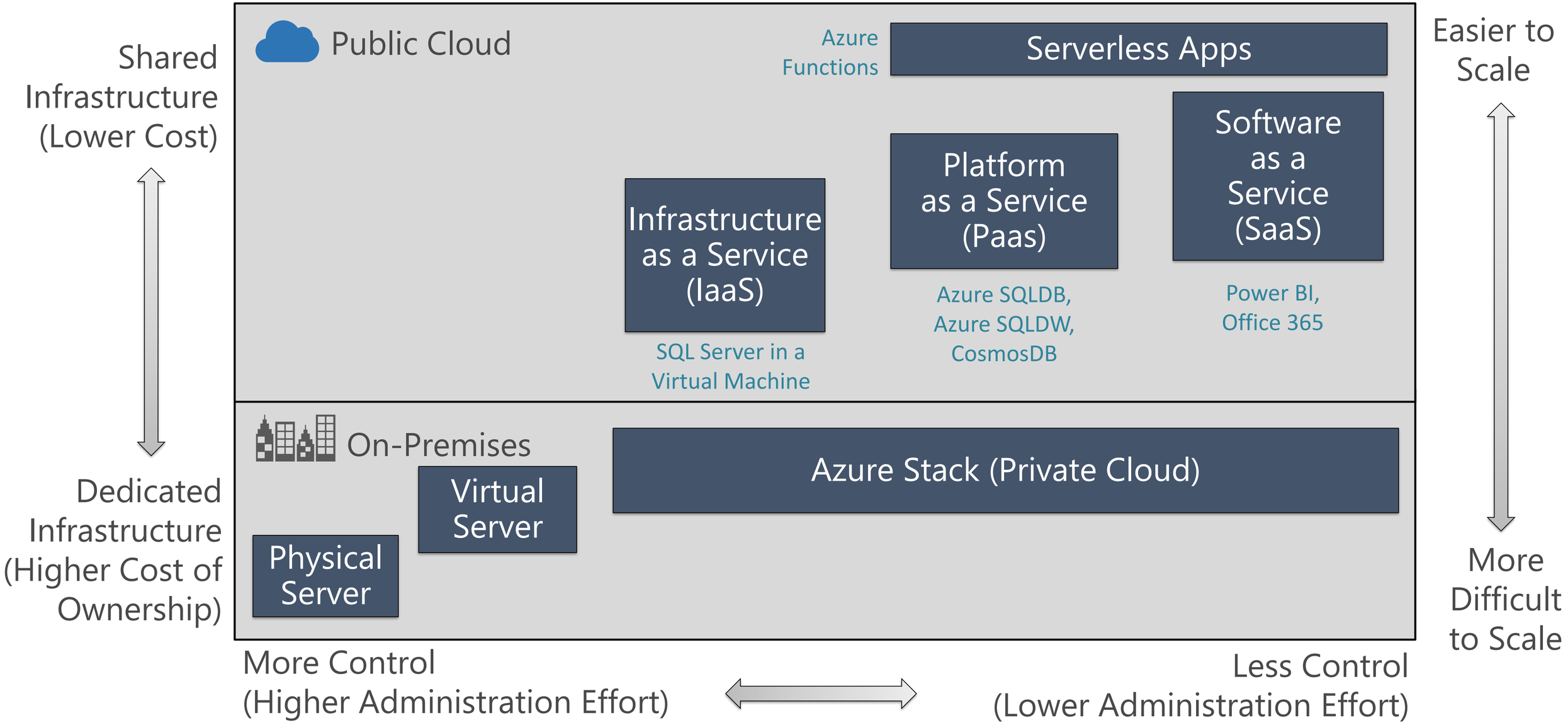
Demo

Find Resources Based on Tags

Tip #6

Determine if IaaS vs PaaS vs SaaS Best Meets Your Needs

IaaS vs. PaaS vs. SaaS



Comparing the SQL Offerings in Azure (1/2)

SQL Server in a Virtual Machine (IaaS)

Run full workload within an Azure virtual machine, including SQL Server, SSIS, SSAS, SSRS, etc

Azure SQL Database (PaaS)

A relational database-as-a-service (DBaaS)

Non-Managed

A traditional Azure SQLDB deployment (isolated DB)

Managed Instance

Newer - closer feature parity to SQL Server (instance level features)

Elastic Pools

Unpredictable multi-tenant OLTP workloads (NOT for data warehousing workloads)

Azure SQL Data Warehouse (PaaS)

An data warehouse-as-a-service (DWaaS) optimized for performance of large-scale analytical workloads

MPP architecture (massively parallel processing)

Comparing the SQL Offerings in Azure (2/2)

SQL Server in a Virtual Machine (IaaS)

Best for:

- ✓ Migrating existing solutions
- ✓ Running all SQL Server features (ex: SSIS, SSRS, SSAS, MDS)
- ✓ Running 3rd party software (ex: Oracle)
- ✓ Bring your own license (Software Assurance)
- ✓ Isolated dev/test environments
- ✓ Administering all aspects (highest level of control)

Azure SQL Database (PaaS)

Best for:

- ✓New database solutions
- ✓Migrating existing database solutions as-is (managed instances which are in preview)
- ✓OLTP with scaling & pooling needs (unpredictable workloads)
- ✓DW workloads which are small to medium
- ✓Reduced administration

Azure SQL Data Warehouse (PaaS)

Best for:

- ✓DW with larger data volumes (ideally 1-4TB+ bare min. of 250GB+)
- ✓Dimensional models which can take advantage of CCI
- ✓Ability to scale compute up/down, or pause (elasticity)
- ✓Data Lake Store integration (relational + nonrelational data)
- ✓Reduced administration

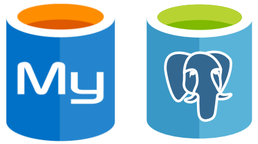
More info: https://docs.microsoft.com/en-us/azure/sql-database/sql-database-paas-vs-sql-server-iaas
https://www.blue-granite.com/blog/is-azure-sql-data-warehouse-a-good-fit

Other PaaS Database Offerings in Azure

Cosmos DB

Best for:

- ✓ Schema-agnostic data model that is write-optimized
  - NoSQL JSON documents
  - Key-value pairs
  - Column-family data
  - Graph data
- ✓ Scalable, globally distributed apps
- ✓ Reduced administration



Azure Database for MySQL and PostgreSQL

Best for:

- ✓Open source DB (community editions)
- ✓Focus is mostly on a front-end application
- ✓Familiar with MySQL or PostgreSQL
- ✓Reduced administration

Azure Analysis Services

Best for:

- ✓Semantic layer for reporting
- ✓Alternative to SQL Server Analysis Tabular Model (otherwise run in an IaaS VM)
- ✓Upsizing a data model from Power BI
- ✓Sharing some workload with Azure SQLDW or Azure SQLDB
- ✓Reduced administration

Azure SQL DB: Feature Parity with SQL Server

Many features go first to Azure SQLDB ("cloud first").

However, some features are *not available* in Azure SQL Database such as:

- PolyBase (alternative: elastic queries)
- SQL Server Machine Learning & R Services
- Change data capture
- CLR
- DB snapshots
- Some T-SQL syntax
- Profiler
- Non-primary filegroups

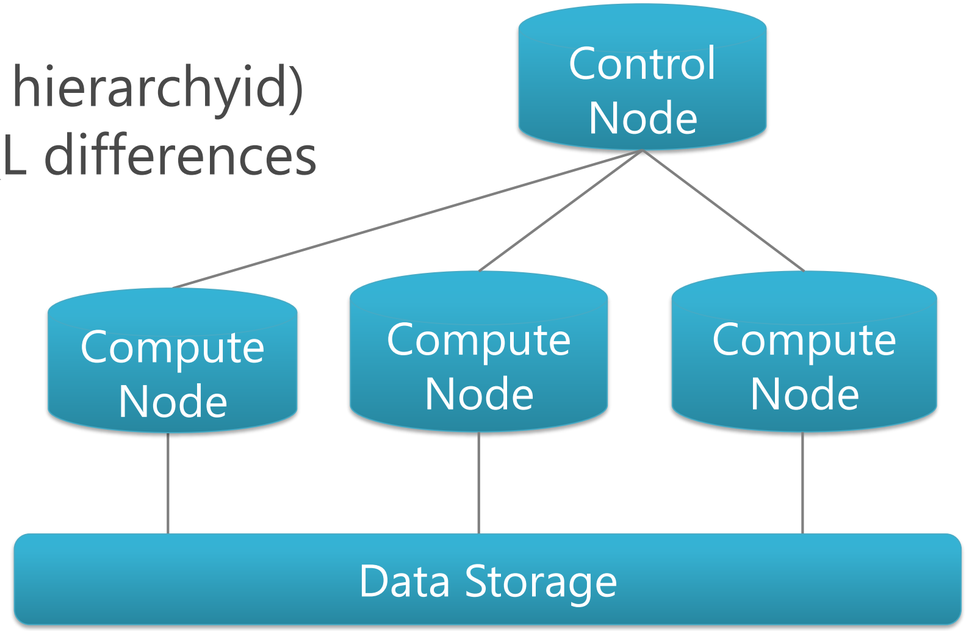
Feature	SQL Server	Azure SQL Database
Active geo-replication	Not supported - see Always On Availability Groups	Supported
Always Encrypted	Supported	Supported - see Cert store and Key vault
AlwaysOn Availability Groups	Supported	Not supported - See active geo-replication
Attach a database	Supported	Not supported
Application roles	Supported	Supported
Auto scale	Not supported	Supported - see Service tiers
Azure Active Directory	Not supported	Supported
Azure Data Factory	Supported	Supported
Aud[illegible]	Supported	Supported

Full list:
https://docs.microsoft.com/en-us/azure/sql-database/sql-database-features

Azure SQL DW: Feature Parity with SQL Server

Some features are *not available* in Azure SQL Data Warehouse:

- Primary key, foreign key, unique key
- Computed columns, default values
- Merge statements
- Some data types (ex: xml, json, geography, geometry, hierarchyid)
- Certain update and delete techniques and other T-SQL differences
- Statistics creation and management
- SSDT database project
- Cross-database queries



The recommended data load technique is via PolyBase.

More info:
https://docs.microsoft.com/en-us/azure/sql-data-warehouse/sql-data-warehouse-troubleshoot#differences-from-sql-database
https://www.blue-granite.com/blog/is-azure-sql-data-warehouse-a-good-fit

Tip #7

Get an Azure SLA if Possible

How to Get a Service Level Agreement (SLA)

SQL Server in a Virtual Machine (IaaS)

SLA on the VM if:

1. Include the VM in an Availability Set - Guarantee: 99.95%
   or
2. Use Premium Storage for all Data and O/S disks - Guarantee: 99.9%

Azure SQL Database (PaaS)

SLA on the database:

✓Guarantee: 99.99% for Basic, Standard, and Premium tiers that customers will have connectivity

Azure SQL Data Warehouse (PaaS)

SLA on the database:

✓Guarantee: 99.99% for Basic, Standard, and Premium tiers that client operations will succeed

Premium Storage is how you get an SLA on a 'single instance' VM. You might consider Standard storage in Dev/Test, and Premium in Prod.

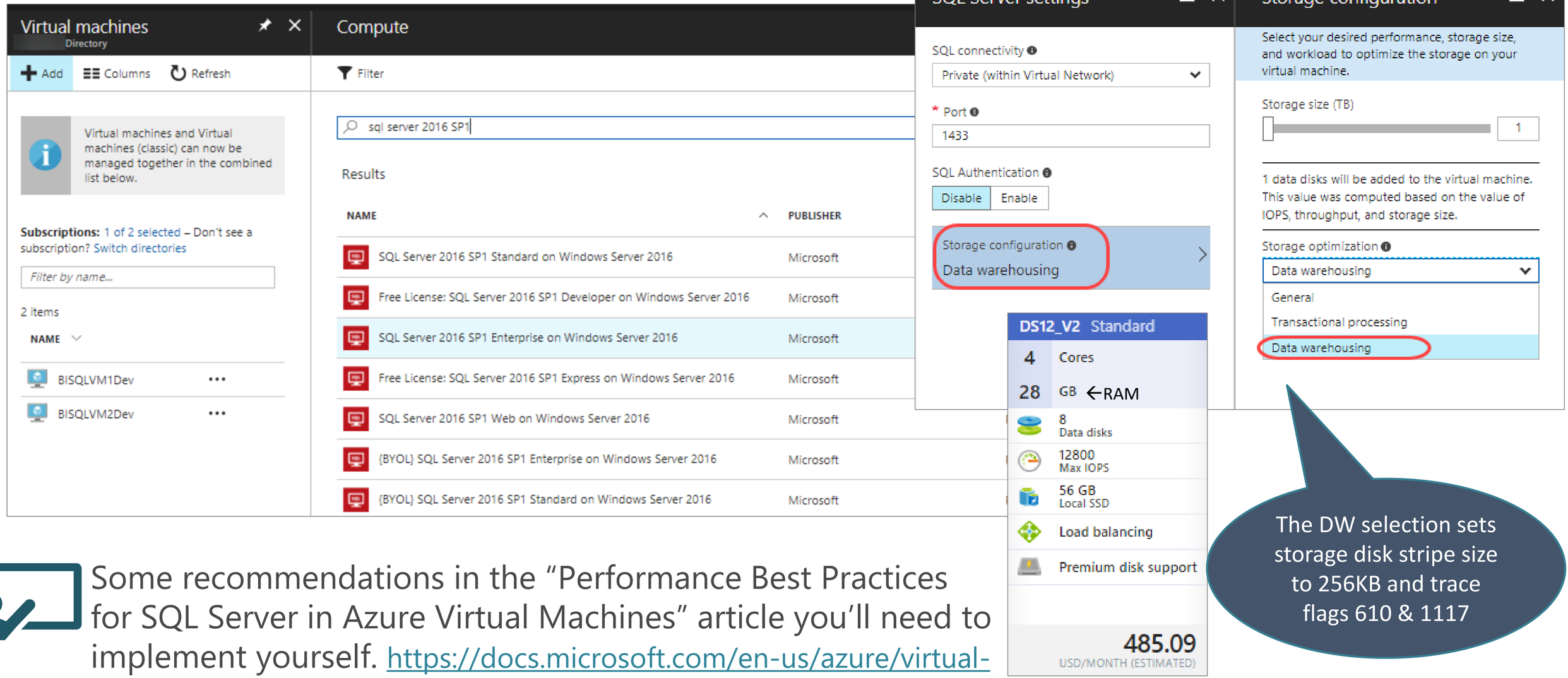
More info: https://azure.microsoft.com/en-us/support/legal/sla/summary/
https://azure.microsoft.com/en-us/support/legal/sla/virtual-machines/v1_6/

Tip #8

Take Advantage of Efficiencies Offered by Cloud Services

Starting With a VM Image

(1/2)



The DW selection sets storage disk stripe size to 256KB and trace flags 610 & 1117

Some recommendations in the "Performance Best Practices for SQL Server in Azure Virtual Machines" article you'll need to implement yourself. https://docs.microsoft.com/en-us/azure/virtual-machines/windows/sql/virtual-machines-windows-sql-performance

Starting With a VM Image (2/2)

Expect to make changes and customizations to the VM image, such as:

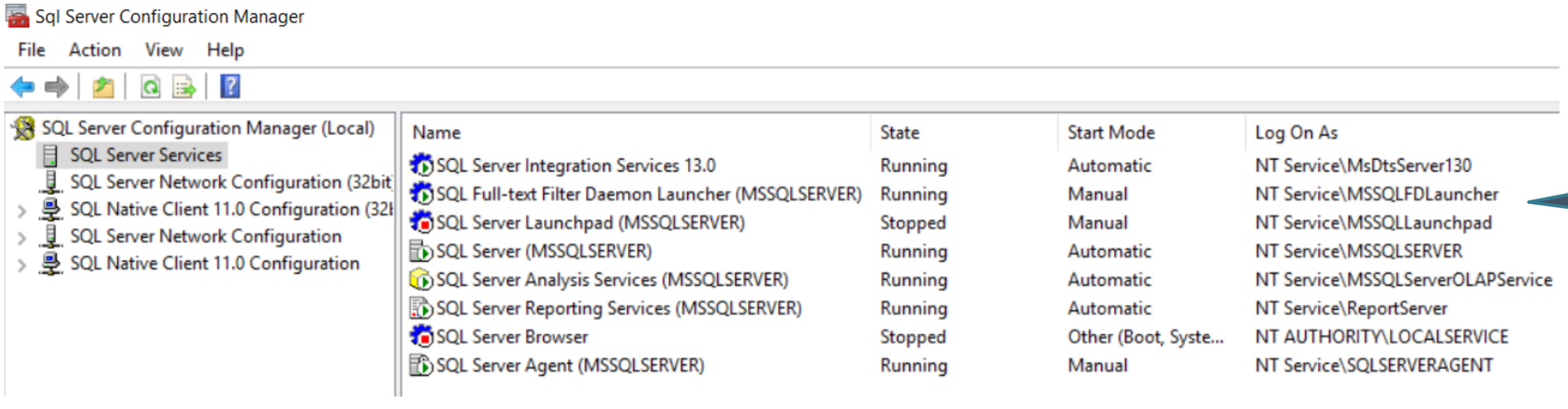
- Domain service accounts for each service
- Disable unused services
- Disk structure + permissions
  - G: Data
  - L: Logs
  - T: TempDB
- Enable disk encryption

By default, there is one data drive on F: provisioned at 1TB. Stay away from D:\ - it's the temp drive.

SQL Server changes to VM image:

- Enable the DAC
- Default DB file locations
- Fill factor
- Optimize for ad hoc workloads
- Max degree of parallelism
- Cost threshold for parallelism
- Startup parameters; trace flags
- Expand # of TempDB files
- Time of SSIS maintenance job

  etc…



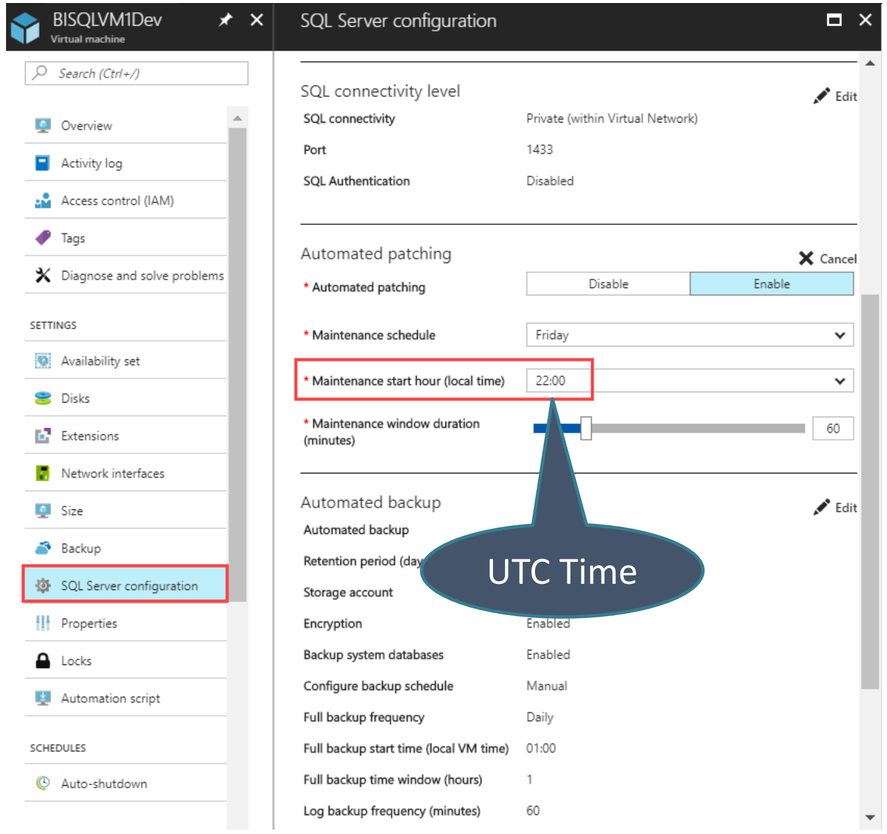
As-is before any customizations

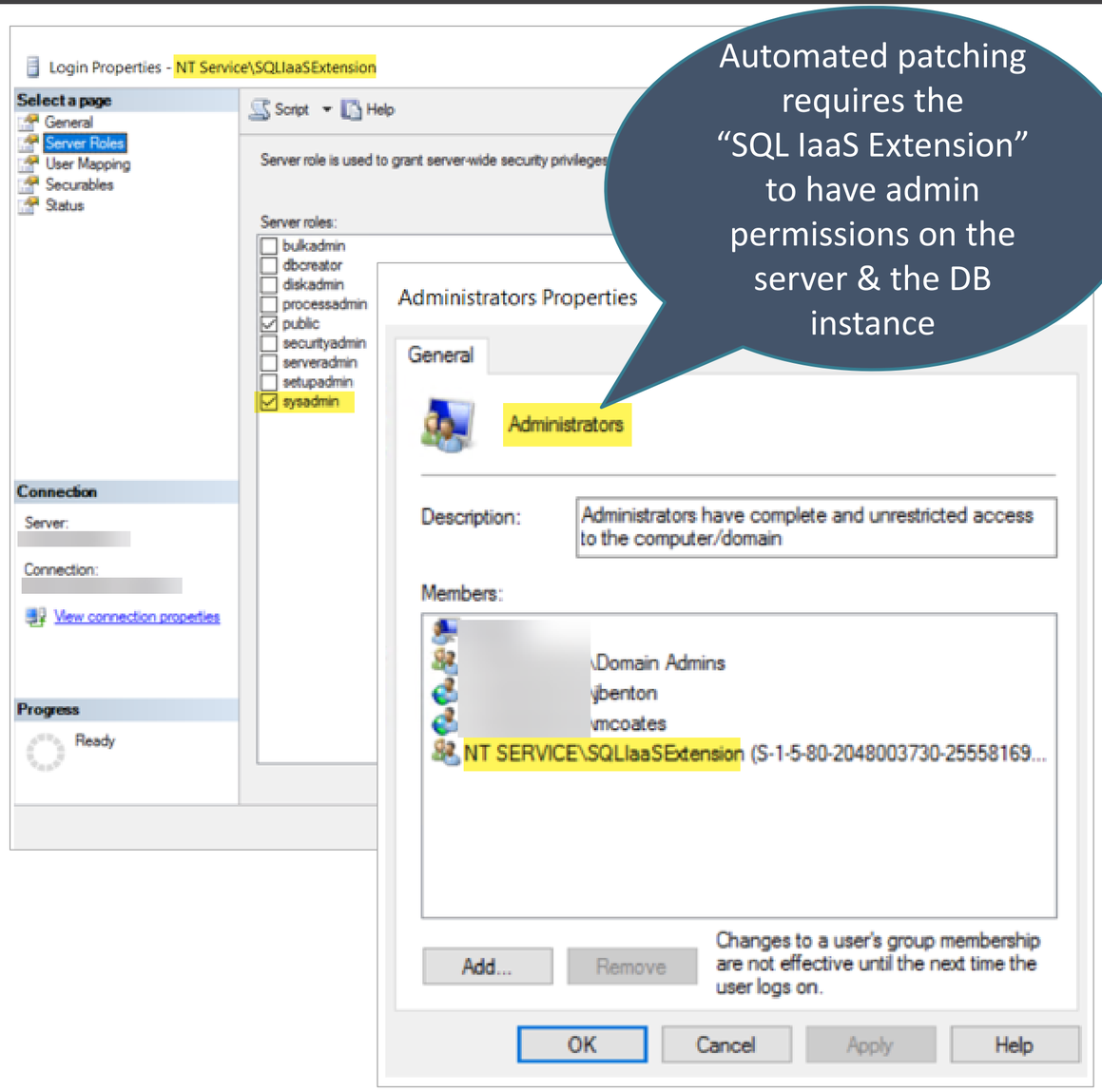
Automatic Patching

Windows and SQL Server patching

PaaS: Done for you (no opt-out or delay).

IaaS: You can optionally allow MSFT to handle. There's also a new VM "Update Management" feature.

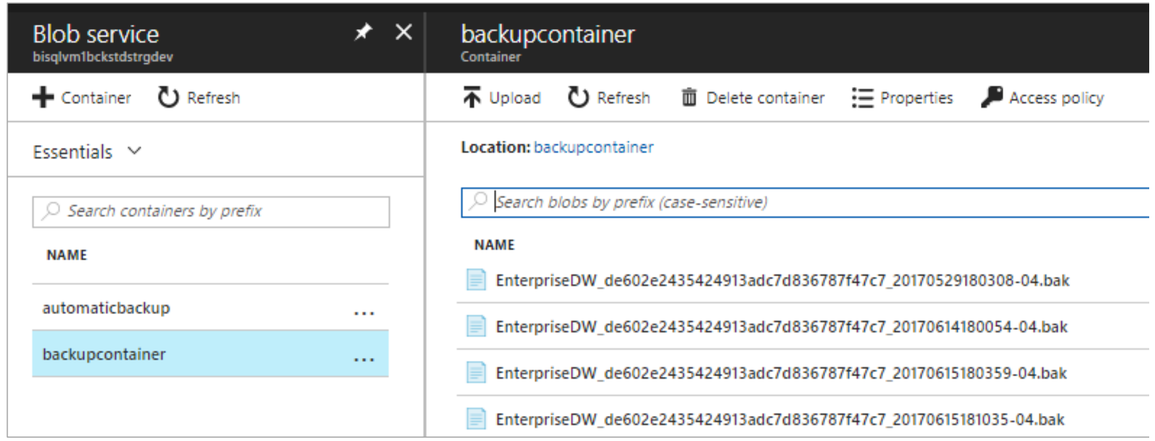




More info: https://docs.microsoft.com/en-us/azure/virtual-machines/windows/sql/virtual-machines-windows-sql-server-agent-extension

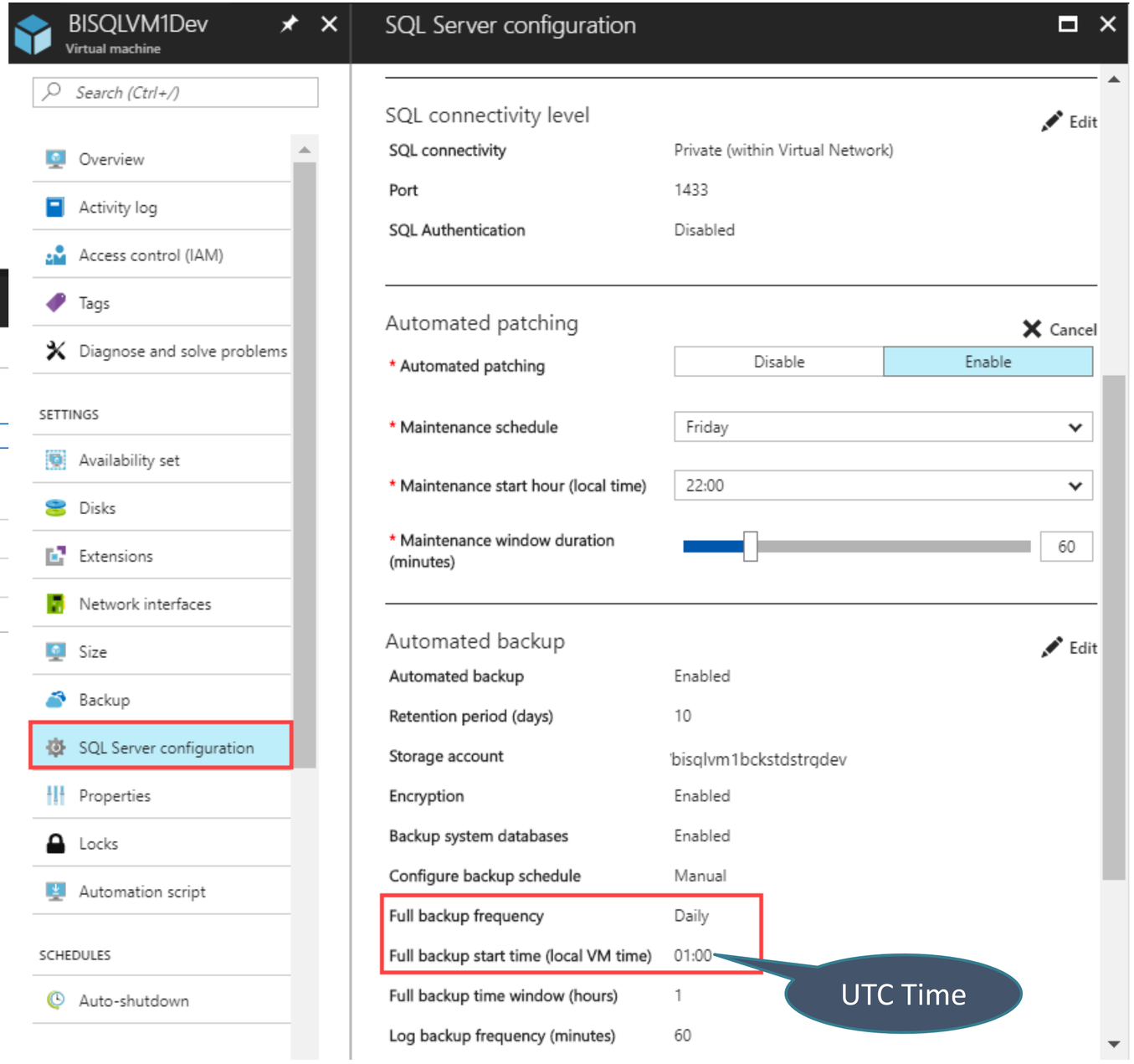
SQL Backups: IaaS

IaaS: You can optionally set up scheduled SQL Server backups.



Suggestions:

- ✓ Be sure that this Azure storage account is set to be encrypted since the "automaticbackup" container contains certificates & keys
- ✓ Provision the storage account with backups in another region, or with geo-redundant storage



SQL Backups: PaaS

Azure SQL Database
(PaaS)

Basic service: retained 7 days
Standard service: retained 35 days
Premium service: retained 35 days

Full backups: weekly
Differential: every few hours
T-log: every 5-10 minutes

Longer-term backup retention:
- Set up a process to back up the .bacpac file (schema + data), or
- Recovery Services vault (still in Preview)

Azure SQL Data Warehouse
(PaaS)

Local snapshots: every 4-8 hours & retained 7 days
Geo-redundant snapshot: retained 24 hours
(Backups occur when DW is online -- not paused)

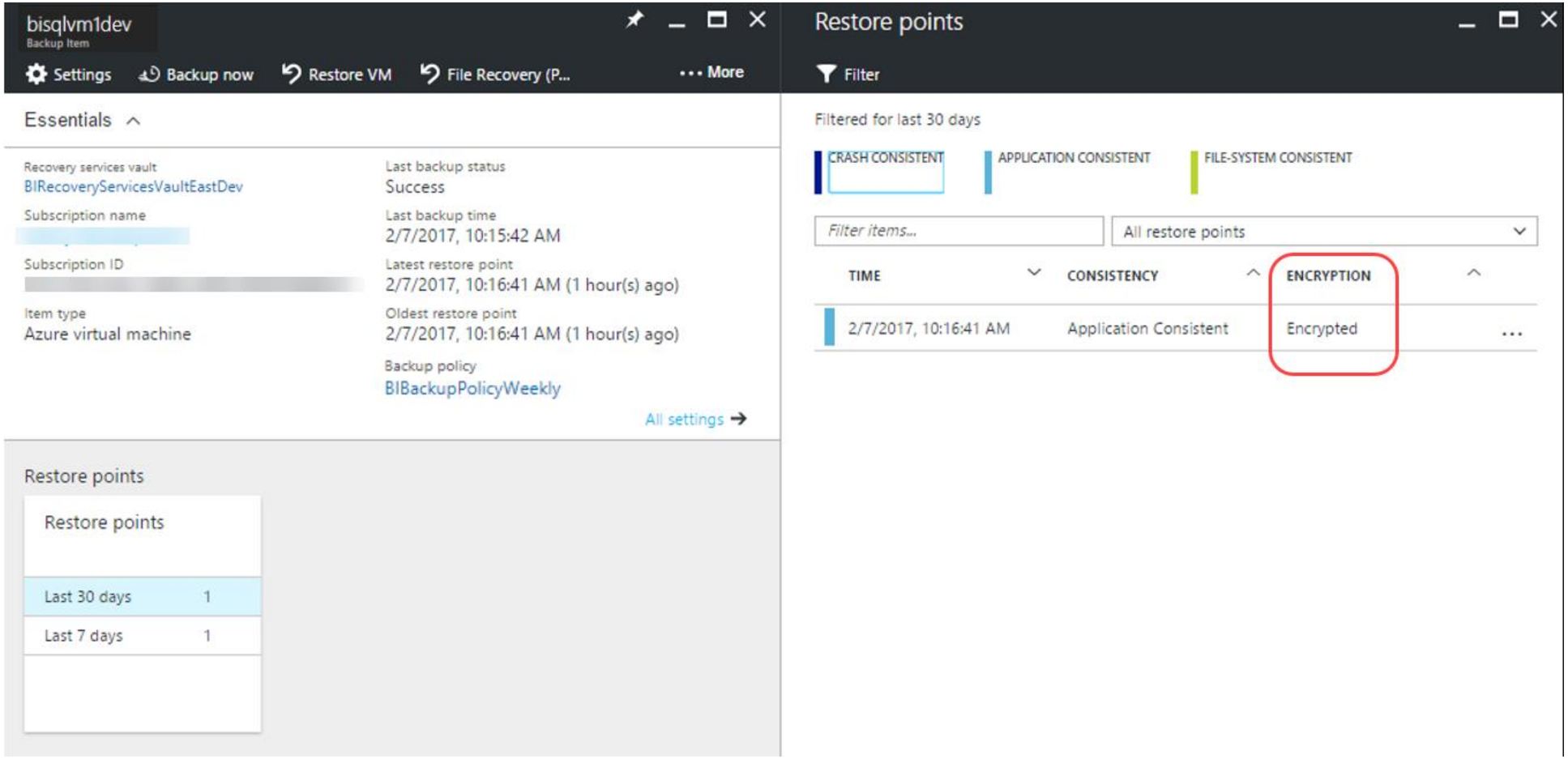
Longer-term backup retention:
- Not currently an option

The tradeoff here: less administration, but also less control

VM Backups

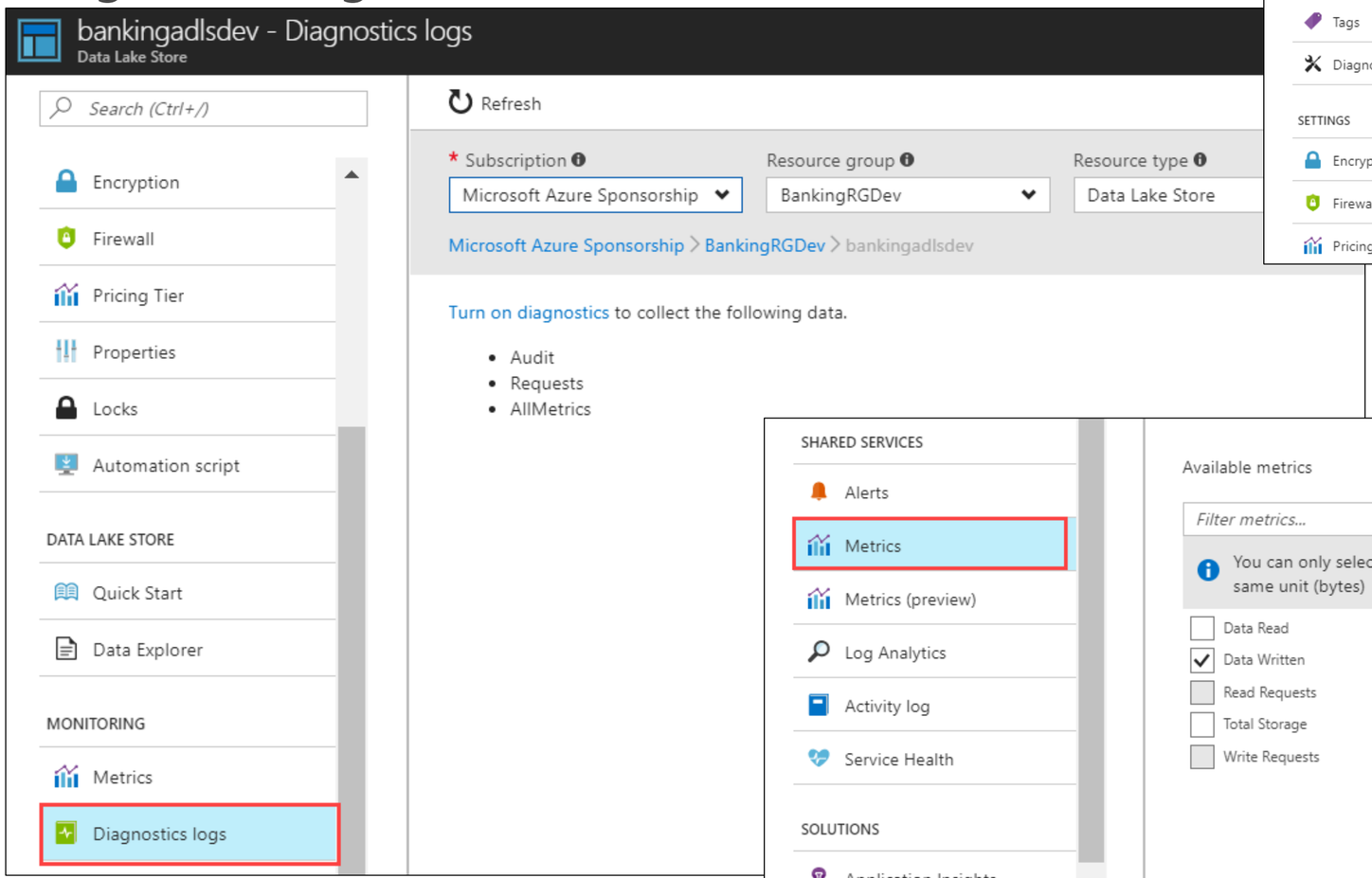
VM backups are managed in the Recovery Services Vault.

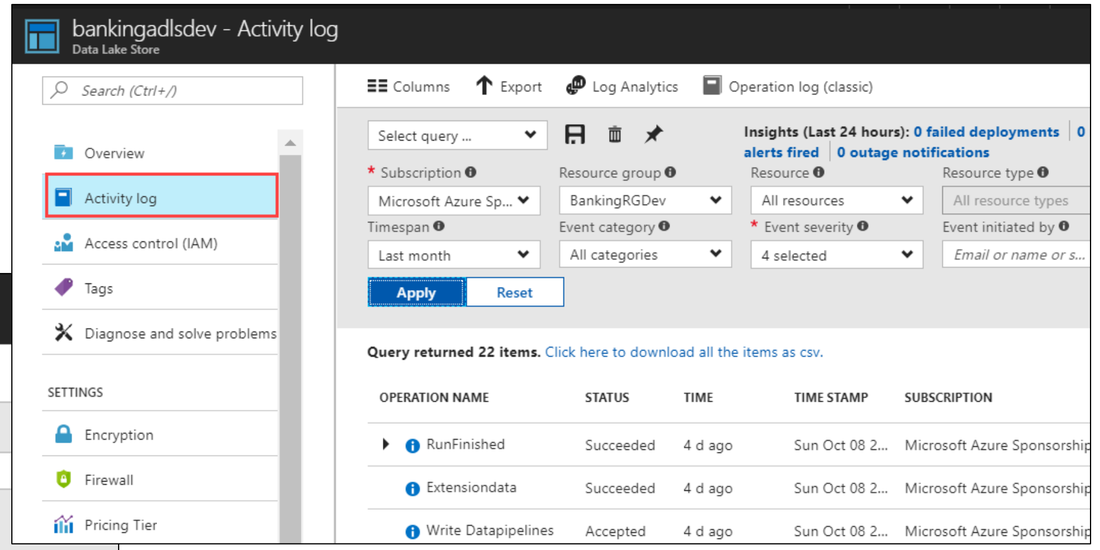
To ensure the VM backups are also encrypted, Azure disk encryption is required.



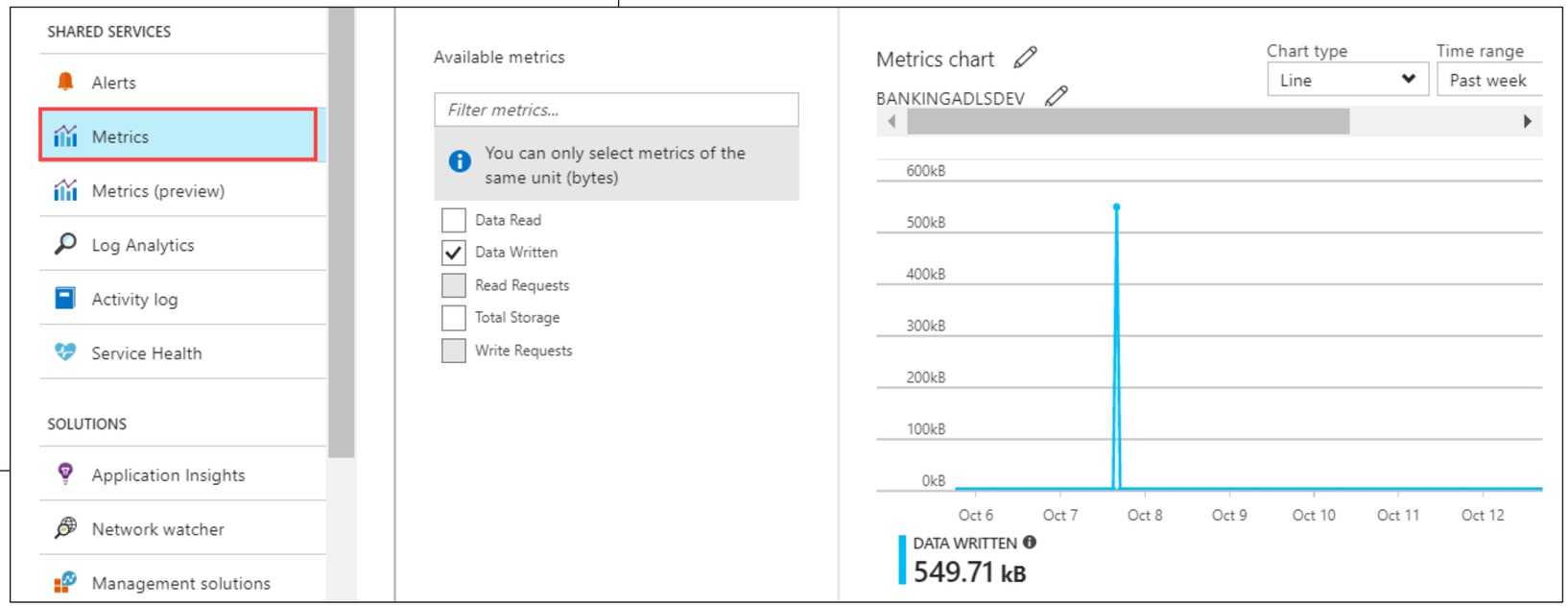
Basic Monitoring

Diagnostic Logs





Activity Log



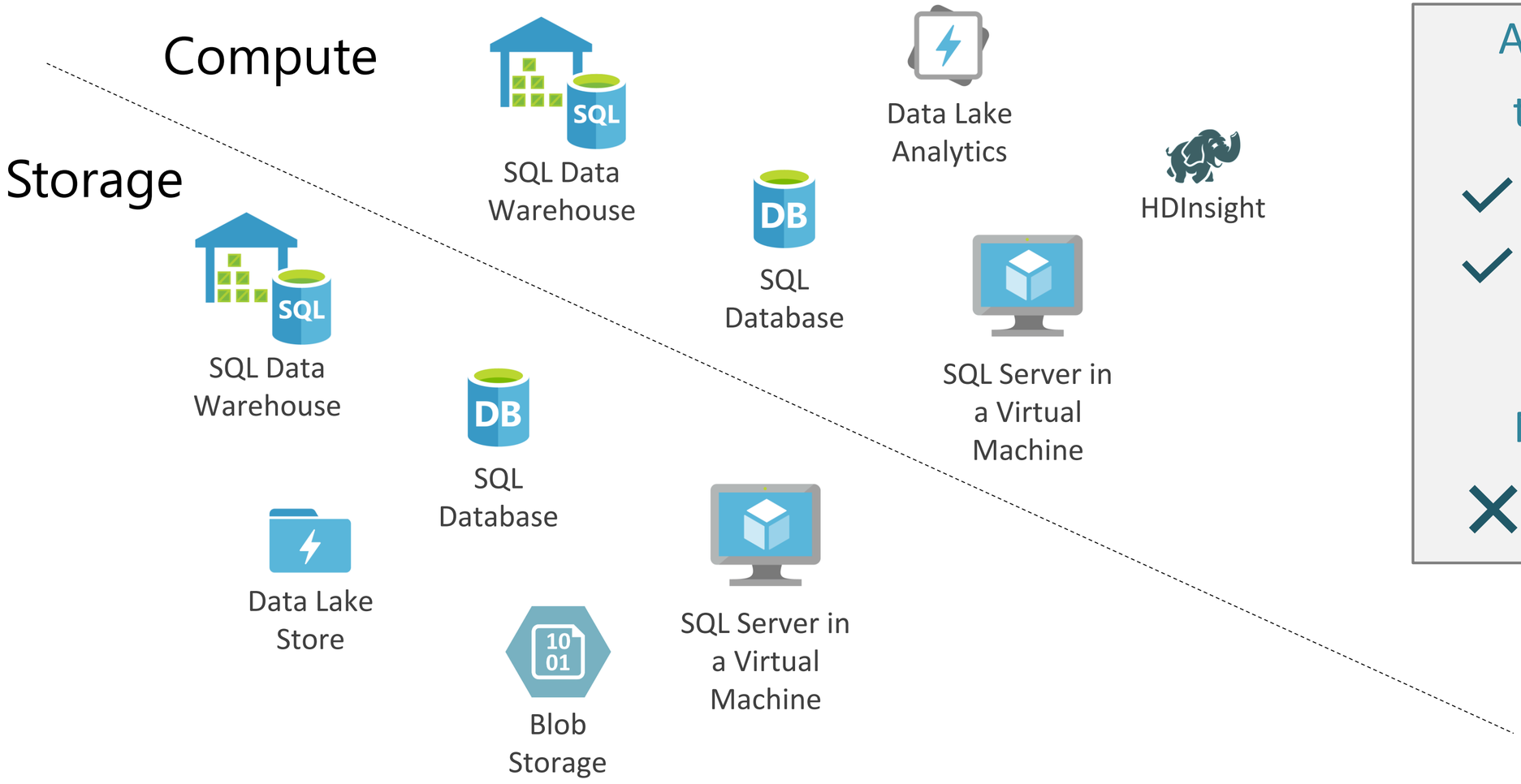
Metrics

Tip #9

Be Aware of Cost Savings Techniques

Scaling and Pausing Compute Resources

Some services support pausing or stopping compute resources when it is idle.
Some services support scaling compute resources up & down to support varying demands.



Ability to Pause/Stop to Save Compute $:

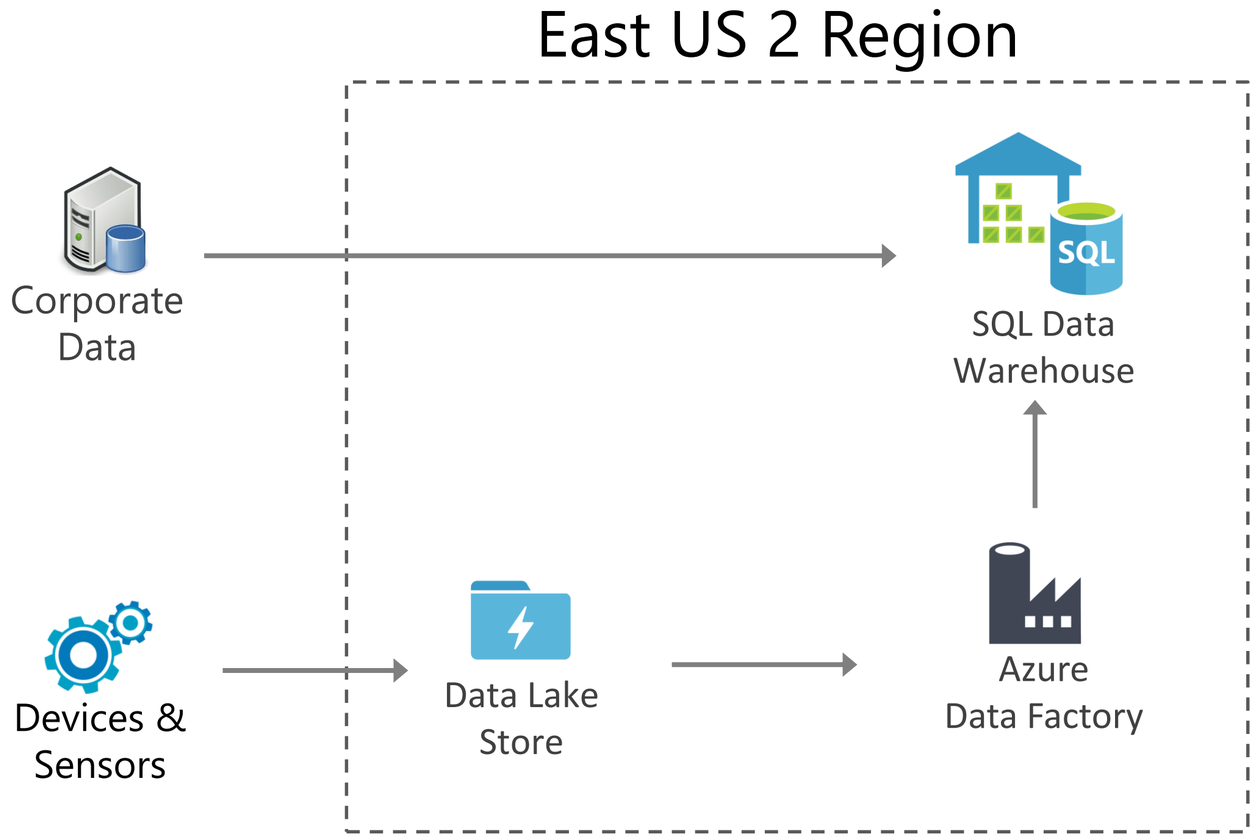
- ✓ Virtual Machine
- ✓ SQL Data Warehouse

No ability to pause:

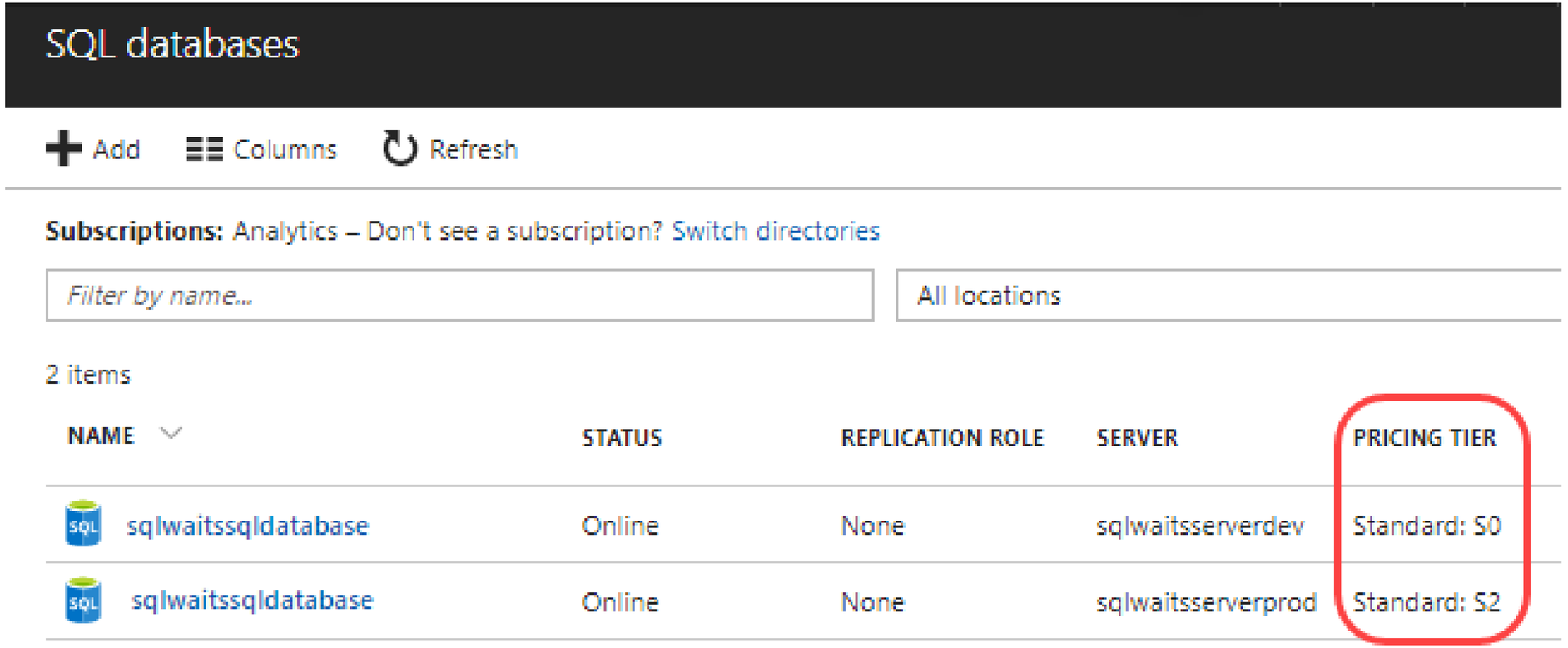
- ✗ SQL Database

Co-Location to Avoid Data Egress Charges

Data egress charges are incurred when data leaves an Azure region. No charges to ingest data into Azure.
Data egress charges are very cheap – but they can easily add up at scale.

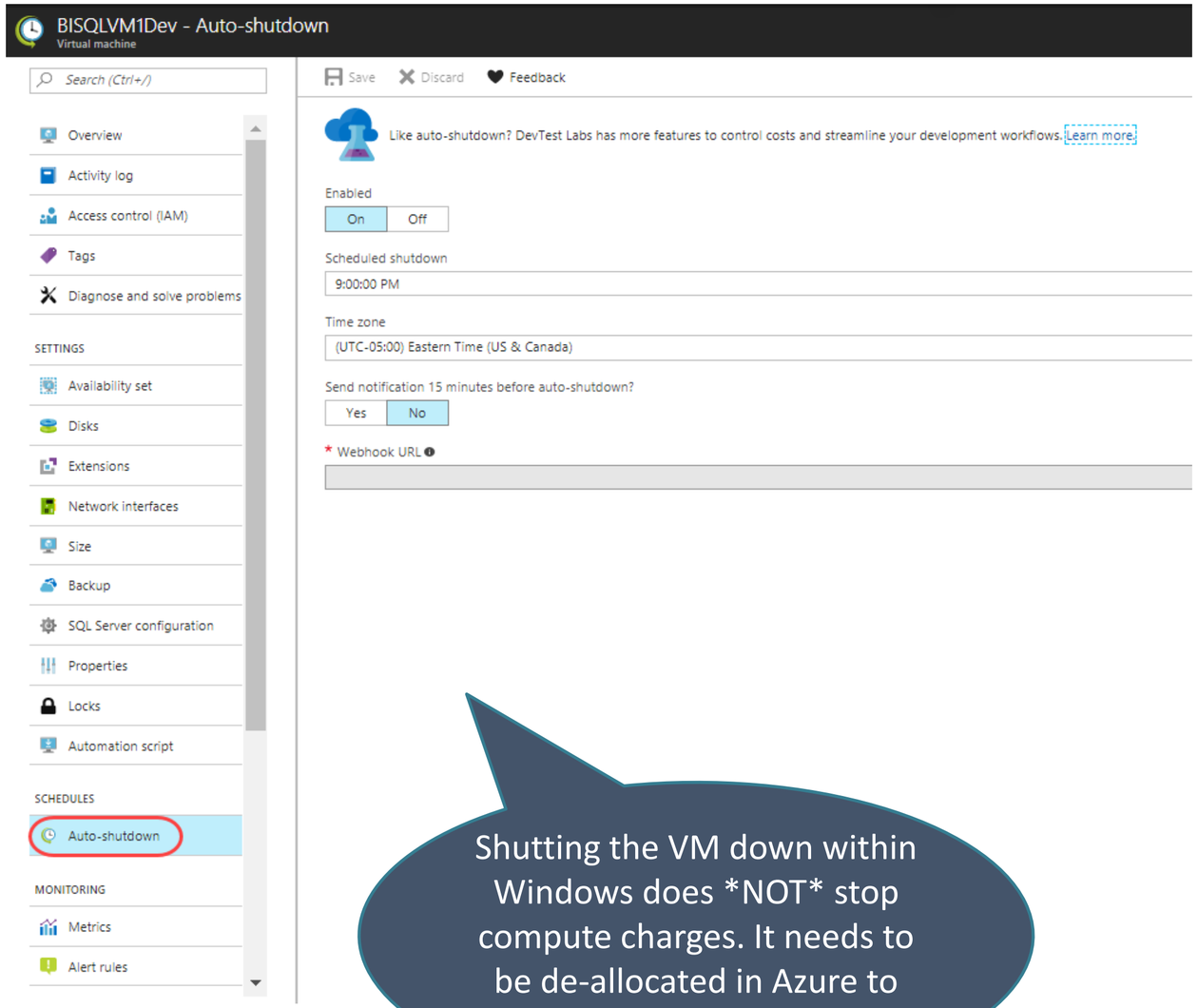


Scale Lower in Dev/Test Environments



In a subscription model, developers and admins can have a huge influence on cost

Automatic Shutdown of VMs



Shutting the VM down within Windows does *NOT* stop compute charges. It needs to be de-allocated in Azure to reduce cost.

Dev & Test

Shutdown:

- ✓ Auto-shutdown feature within the VM properties

Start:

- ✓ VM is manually restarted when needed by one of the developers

Production

Shutdown & Start:

- ✓ A machine which needs to be restarted by a certain time (ex: DB engine+SSIS) can be done with:
  - Azure Automation (PowerShell)
  - Microsoft Flow
  - Azure Functions
  - etc…

Saving Money Running Virtual Machines

Low Priority VMs on Azure Batch

Runs on a pool of VMs; takes advantage of compute surplus in Azure

Useful when:

- ✓ Job completion time is flexible
- ✓ Batch job can be parallelized & distributed across VMs

https://docs.microsoft.com/en-us/azure/batch/batch-low-pri-vms

https://blogs.msdn.microsoft.com/uk_faculty_connection/2017/05/14/microsoft-azure-makes-vms-affordable-for-everybody/

Prepaid Reserved VM

Upfront commitment for compute capacity at a large discount

Useful when:

- ✓ Predictable, long-term workloads

https://azure.microsoft.com/en-us/pricing/reserved-vm-instances/

"Burstable" B-Series VMs

Accumulates 'saved' credits for unused CPU cycles—uses the credits for bursting

Useful when:

- ✓ Workload is small but a dedicated VM is still desirable
- ✓ Bursts only occasionally (ex: once a day)

https://buildazure.com/2017/09/11/more-affordable-azure-vms-with-burstable-b-series/

Dev/Test Labs

Offers lower rates for certain Azure services.

Useful for:
- ✓ Learning
- ✓ Experimenting
- ✓ Proof of concept
- ✓ Throwaway work
- ✓ Non-production

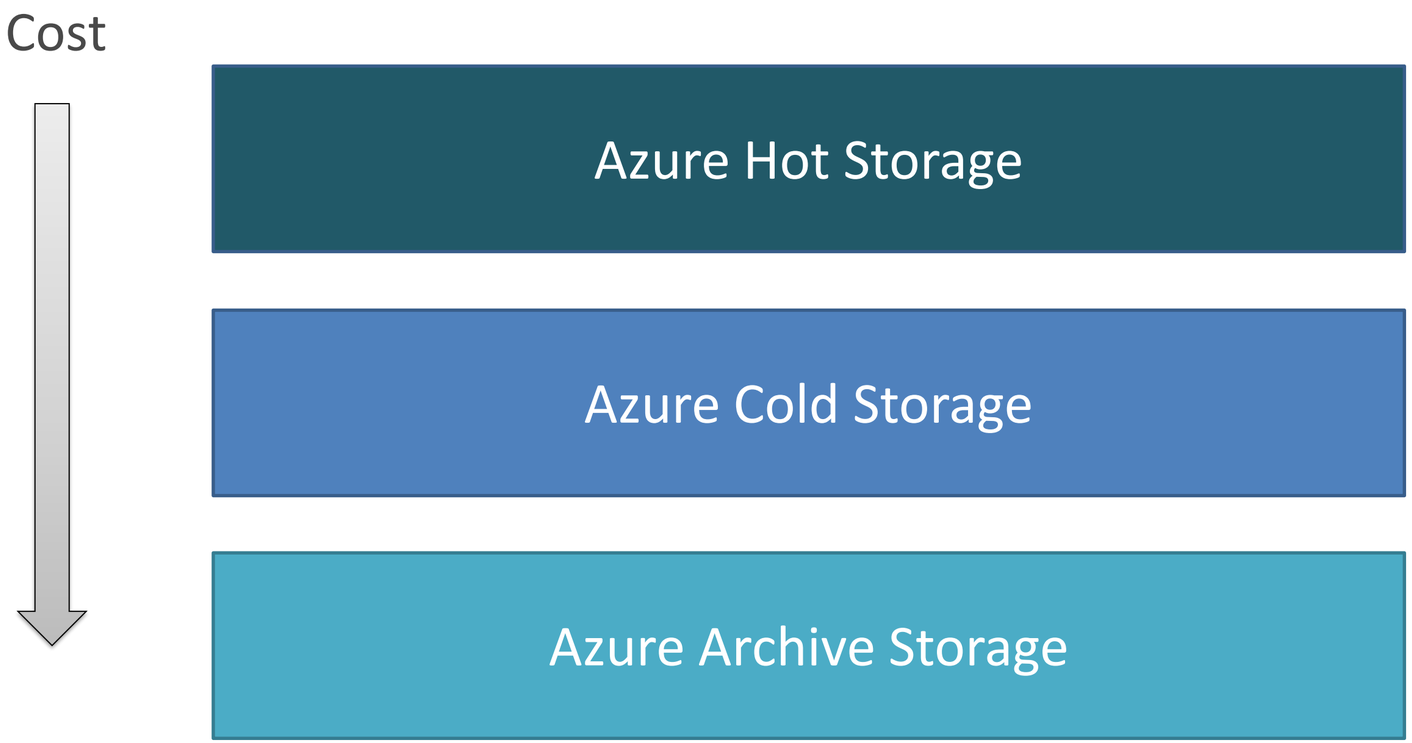
Specify policies and thresholds to control cost.

Requires active Visual Studio subscription.

	FOR INDIVIDUALS	FOR TEAMS	FOR TEAMS
Dev/Test pricing options	Monthly Azure credits for Visual Studio subscribers	Enterprise Dev/Test	Pay-As-You-Go Dev/Test
Number of instances allowed	One per Visual Studio subscriber	Unlimited	Unlimited
Monthly credit included*	$50, $100, or $150	None	None
Billing options	No credit card required Credit card billing for overages (invoicing available)	Always billed to the Enterprise Agreement Consumes prepaid Azure Monetary Commitment funds, if available	Credit card billing (invoicing available)
Billed usage	Overages are billed, after the monthly credit is consumed	All usage is billed	All usage is billed
Allowed to run Windows 10 Virtual Machines	✓	✓	✓
Service Level Agreement	Only for Visual Studio Team Services, Application Insights, and HockeyApp	Only for Visual Studio Team Services, Application Insights, and HockeyApp	Only for Visual Studio Team Services, Application Insights, and HockeyApp
Licensing terms	Restricted to dev/test use only For use by active Visual Studio subscribers, and by end users providing feedback and performing acceptance tests	Restricted to dev/test use only For use by active Visual Studio subscribers, and by end users providing feedback and performing acceptance tests	Restricted to dev/test use only For use by active Visual Studio subscribers, and by end users providing feedback and performing acceptance tests

https://azure.microsoft.com/en-us/pricing/dev-test/
https://azure.microsoft.com/en-us/services/devtest-lab/

Efficient Use of Storage Tiers



Data which is accessed frequently

Data which is accessed less frequently

Data which is not accessed

https://docs.microsoft.com/en-us/azure/storage/blobs/storage-blob-storage-tiers

Virtual Machine Storage

	Premium Disk	Standard Disk
Managed Disk	Billed for disk size reservation	Billed for disk size reservation
Unmanaged Disk	Billed for disk size reservation	Billed for space used

Premium (SSD)

IOPS & throughput are associated with disk size

	P4	P6	P10	P20	P30	P40	P50
Disk Size	32 GB	64 GB	128 GB	512 GB	1 TB	2 TB	4 TB
Price per month	$4.81	$9.29	$17.92	$66.56	$122.88	$235.52	$450.56
IOPs per disk	120	240	500	2,300	5,000	7,500	7,500
Throughput per disk	25 MB/second	50 MB/second	100 MB/second	150 MB/second	200 MB/second	250 MB/second	250 MB/second

Standard (HDD)

IOPS & throughput are based on VM size + tier + # of disks

	S4	S6	S10	S15	S20	S30	S40	S50
Disk Size	32 GB	64 GB	128 GB	256 GB	512 GB	1 TB	2 TB	4 TB
Price per month (promotional)	$0.77	$1.51	$2.95	$5.67	$10.88			

VM Tier	Basic Tier VM	Standard Tier VM
Max Disk size	4095 GB	4095 GB
Max 8 KB IOPS per disk	Up to 300	Up to 500
Max Bandwidth per disk	Up to 60 MB/s	Up to 60 MB/s

Use Scripts to Override Default Storage Sizes

Azure SQL DW

- ✓ Requires Premium storage
- ✓ Provisions 1 TB by default if initially created in the portal (or if maxsize is unspecified in a script)

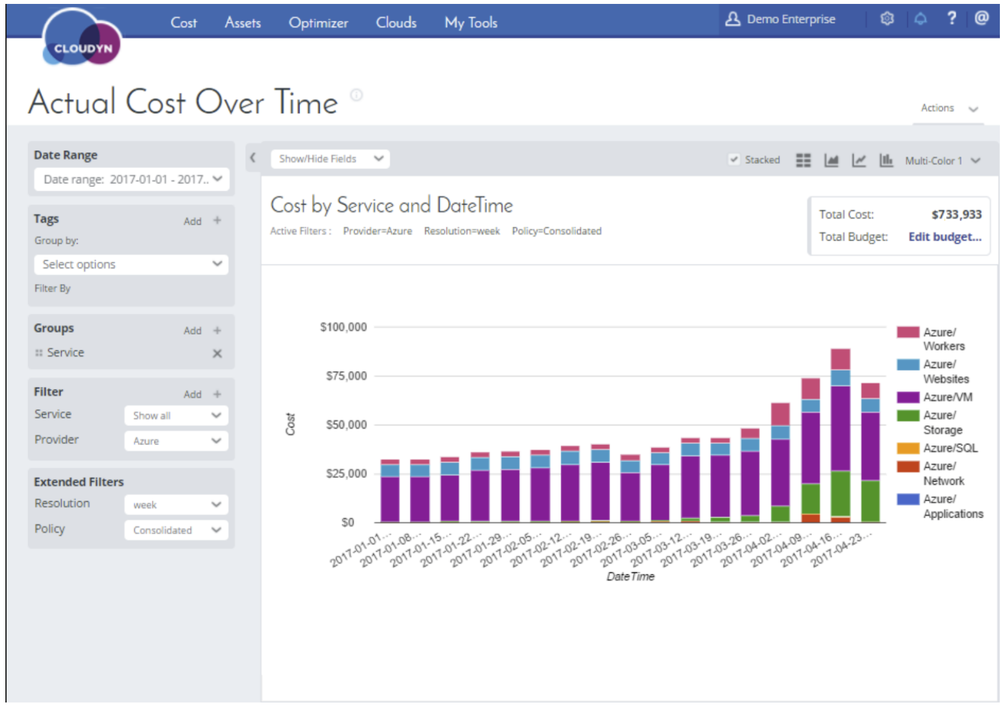
```
CREATE DATABASE BankingSQLSQLDW
COLLATE SQL_Latin1_General_CP1_CI_AS
(
    MAXSIZE = 250 GB
  ,EDITION = 'datawarehouse'
  ,SERVICE_OBJECTIVE = 'DW100'
);
```

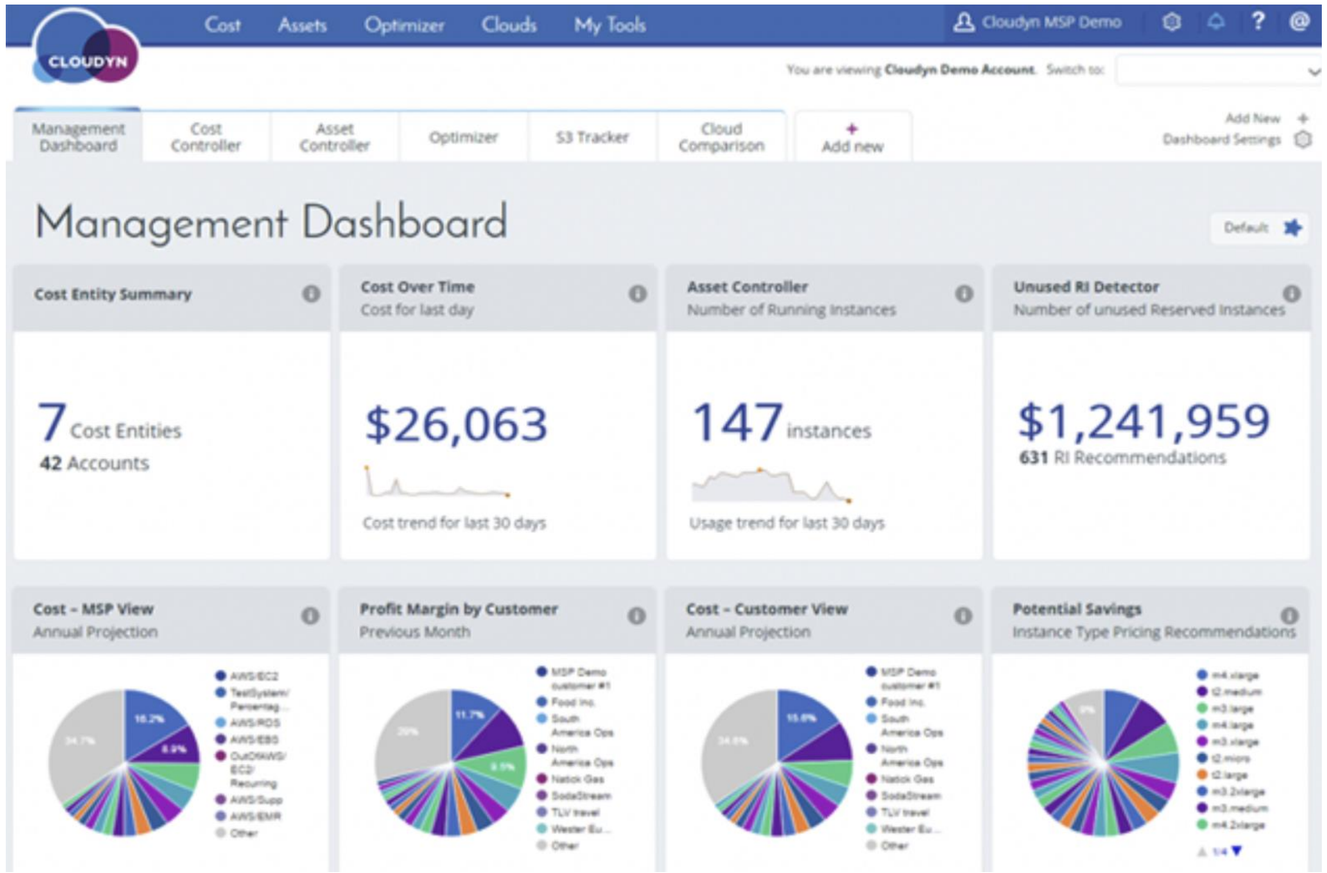
Virtual Machine

- ✓ Pre-built images provision 1 TB by default

Azure Cost Management

A service offering in Azure from Cloudyn to monitor & manage cloud spend. Basic tier is free.

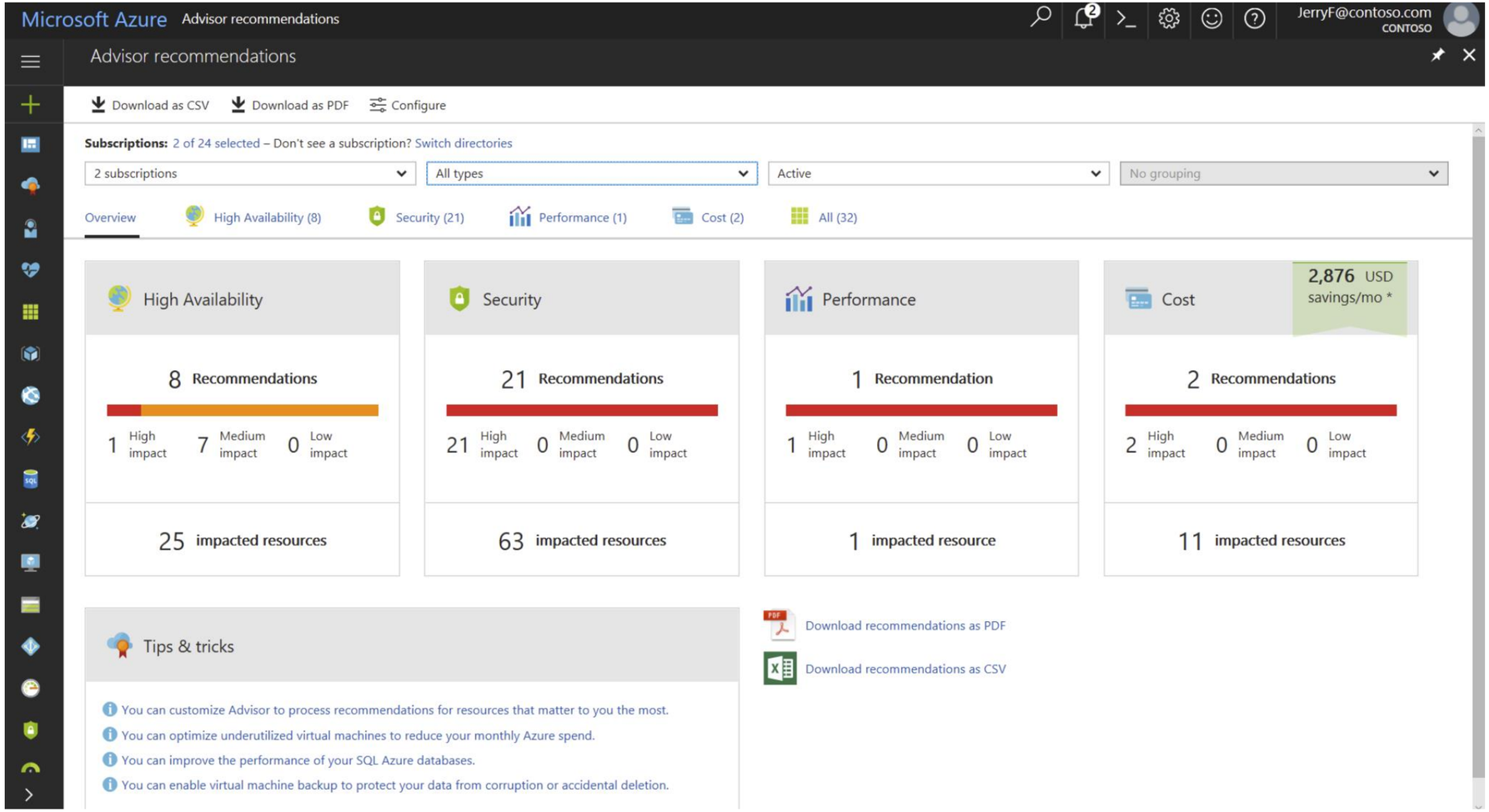




https://docs.microsoft.com/en-us/azure/cost-management/

Azure Advisor

A ‘best practices’ advisor to optimize Azure deployments, including cost optimizations.



https://docs.microsoft.com/en-us/azure/advisor/advisor-overview

Demo

Auto-VM Shutdown + SQLDW Pause + SQLDB Scale

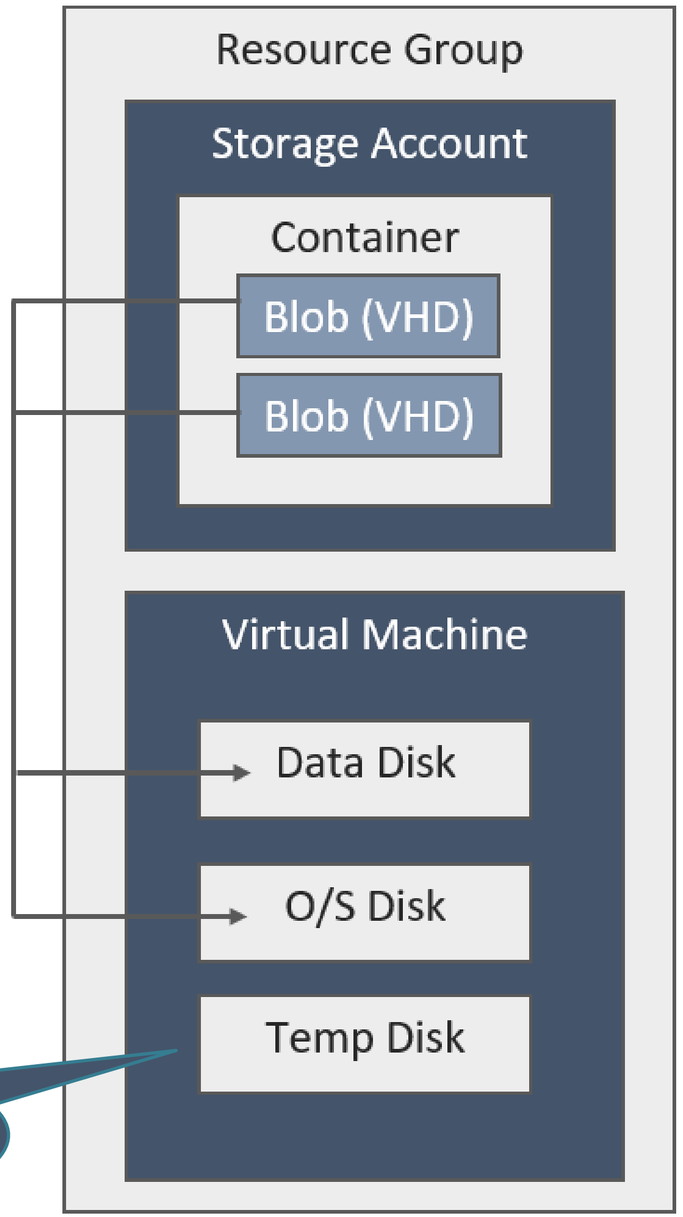
Tip #10

Understand Implications of Storage Decisions for IaaS VMs

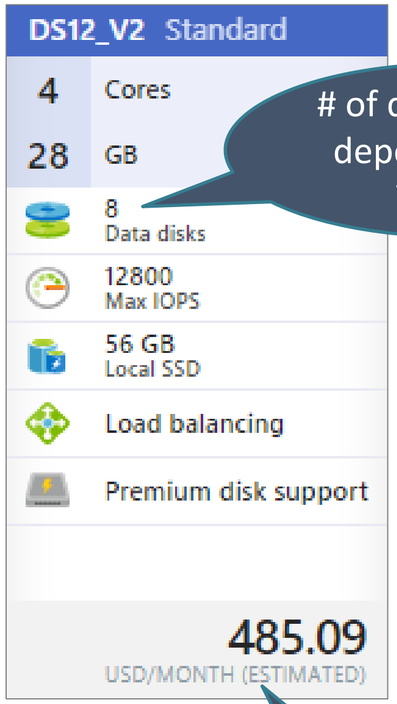
Storage for VMs

The VHDs (virtual hard disks) behind a VM are in Azure Storage

If you delete a VM in Azure, don't forget to also go delete the related storage (if appropriate)



Don't use the temp disk for anything persistent



# of disks allowed depends on the VM size

The VM estimate doesn't include the O/S disk or data disk(s)

Storage Decisions Related to VMs

1. Managed or Unmanaged
2. Premium (SSD) or Standard (HDD)
3. Redundancy (Local, Zone, or Geo-Redundant)
4. Encryption (Storage Service Encryption and/or Disk Encryption)

Local is recommended per the Best Practices article

	Premium Disk	Standard Disk
Managed Disk	Billed for disk size reservation	Billed for disk size reservation
Unmanaged Disk	Billed for disk size reservation	Billed for space used

Can’t currently reduce a disk size reservation – though it can be increased easily

Organizing Unmanaged Storage for VMs

All data for a single VM is in the same storage account – it is a unit of recovery in case of failure

NAME	RESOURCE GROUP	LOCATION	SKU	BILLINGCATEGORY (TAG)	ENVIRONMENTTYPE (TAG)	SUPPORTCONTACT (TAG)
INTERNALREPORTINGRGDEV						
bisqlvm1bckstdstrgdev	InternalReportingRGDev	East US	Standard_RAGRS	Internal Analytics	Dev	BI & Analytics Team
bisqlvm1datastdstrgdev	InternalReportingRGDev	East US	Standard_LRS	Internal Analytics	Dev	BI & Analytics Team
bisqlvm1diagstdstrgdev	InternalReportingRGDev	East US	Standard_LRS	Internal Analytics	Dev	BI & Analytics Team
SQLSKILLSWAITSLIBRARYRGDEV						
sqlwaitsbckstdstrgdev	SQLSkillsWaitsLibraryRGDev	East US	Standard_RAGRS	SQLSkills Waits Library	Dev	BI & Analytics Team
sqlwaitsdiagstdstrgdev	SQLSkillsWaitsLibraryRGDev	East US	Standard_LRS	SQLSkills Waits Library	Dev	BI & Analytics Team

Backups (“bck”), Data, and Diagnostic logging (“diag”) are separated

Read Access Geo-Redundancy (RAGRS) for backup files

VM Disk Performance

Premium (SSD) – IOPS & Throughput are Associated with Disk Size:

	P4	P6	P10	P20	P30	P40	P50
Disk Size	32 GB	64 GB	128 GB	512 GB	1 TB	2 TB	4 TB
Price per month	$4.81	$9.29	$17.92	$66.56	$122.88	$235.52	$450.56
IOPs per disk	120	240	500	2,300	5,000	7,500	7,500
Throughput per disk	25 MB/second	50 MB/second	100 MB/second	150 MB/second	200 MB/second	250 MB/second	250 MB/second

Standard (HDD) – IOPS & Throughput are Based on VM Size + Tier + # of Disks:

	S4	S6	S10	S15	S20	S30	S40	S50
Disk Size	32 GB	64 GB	128 GB	256 GB	512 GB	1 TB	2 TB	4 TB
Price per month (promotional)	$0.77	$1.51	$2.95	$5.67	$10.88	$20.48	$40.96	$81.92

VM Tier	Basic Tier VM	Standard Tier VM
Max Disk size	4095 GB	4095 GB
Max 8 KB IOPS per disk	Up to 300	Up to 500
Max Bandwidth per disk	Up to 60 MB/s	Up to 60 MB/s

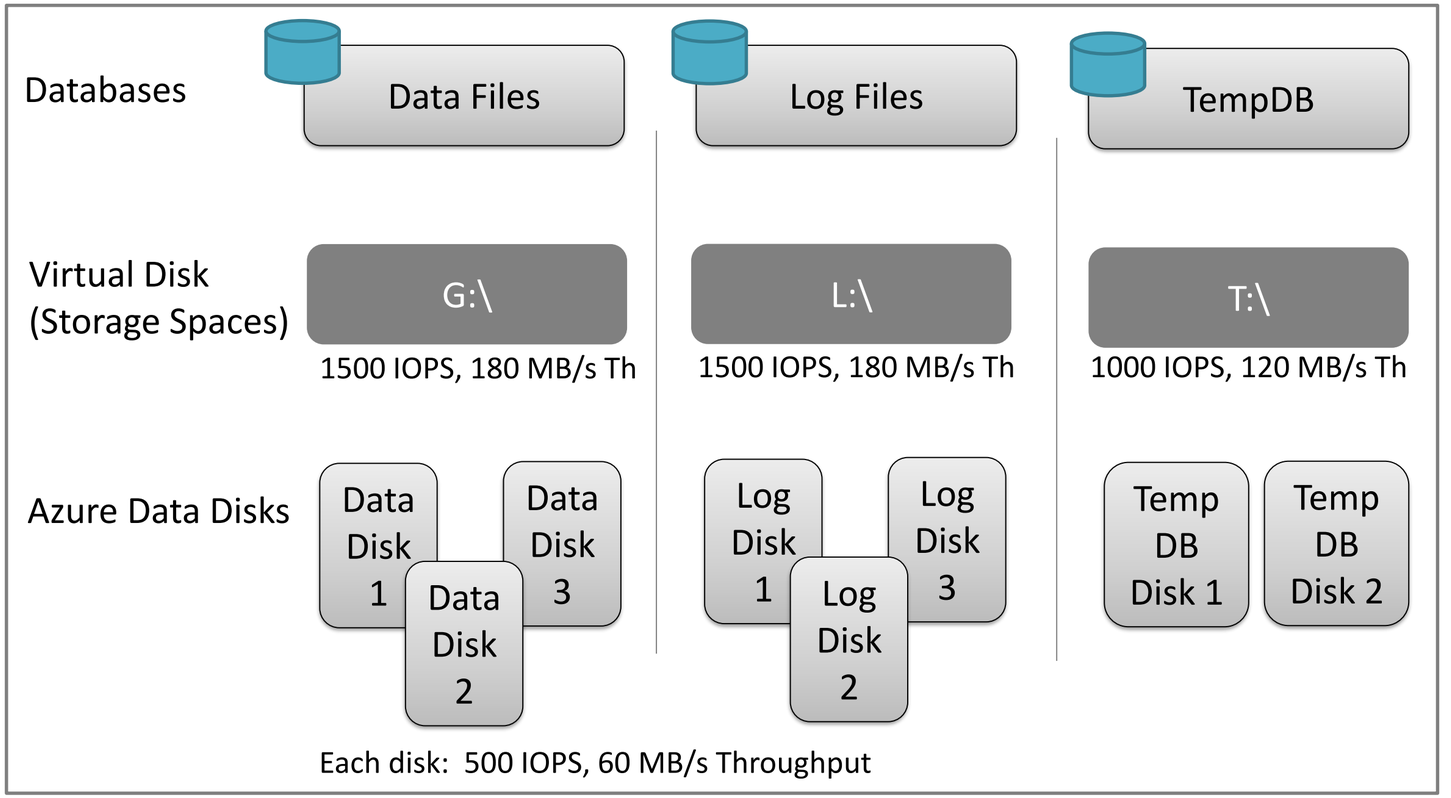
https://docs.microsoft.com/en-us/azure/storage/common/storage-premium-storage
https://docs.microsoft.com/en-us/azure/storage/common/storage-standard-storage

Increasing Performance of Standard Disks (1/3)

Storage Spaces (aka Virtual Disks) are multiple Azure data disks in a storage pool to collectively share throughput and IOPS limits.

This aggregation of disks is helpful due to lower Standard limits on IOPS & throughput.

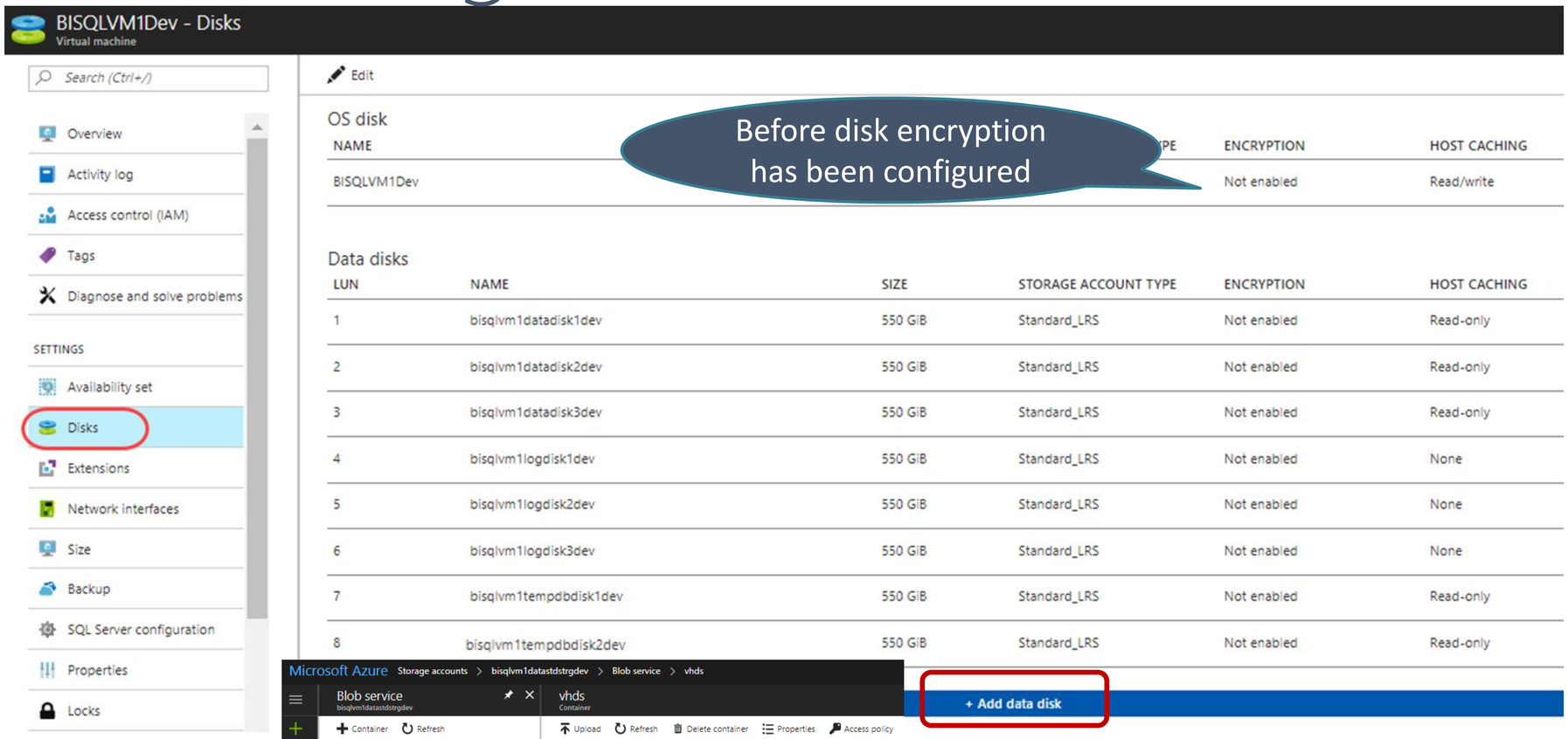
(Another alternative to increase disk performance is to spread separate database files on different disks.)



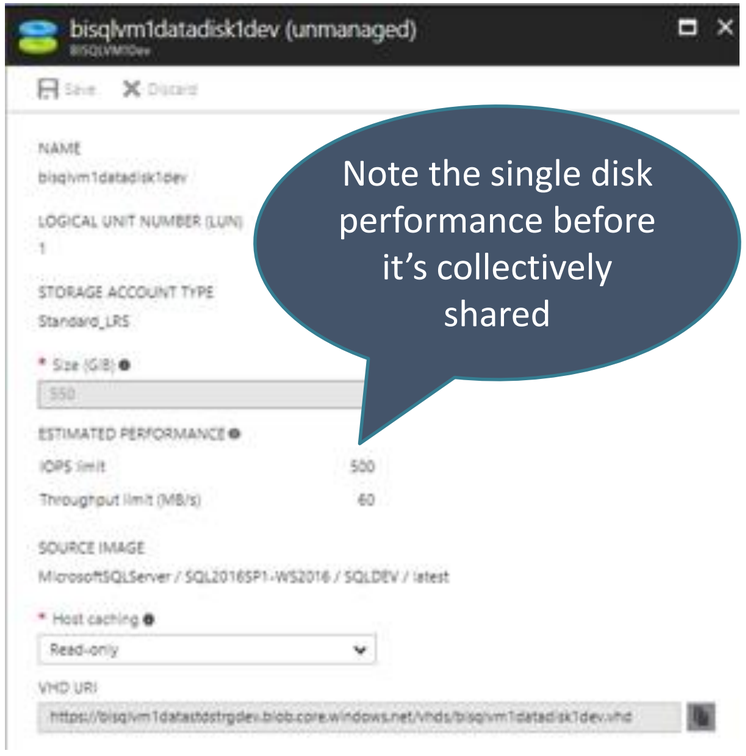
Adapted from:
https://blogs.msdn.microsoft.com/sqlcat/2013/06/17/performance-guidance-for-sql-server-in-windows-azure-virtual-machines/

IOPS = # of requests per second (OLTP workloads)
Throughput (Bandwidth) = IOPS x I/O size per specified interval (DW workloads)

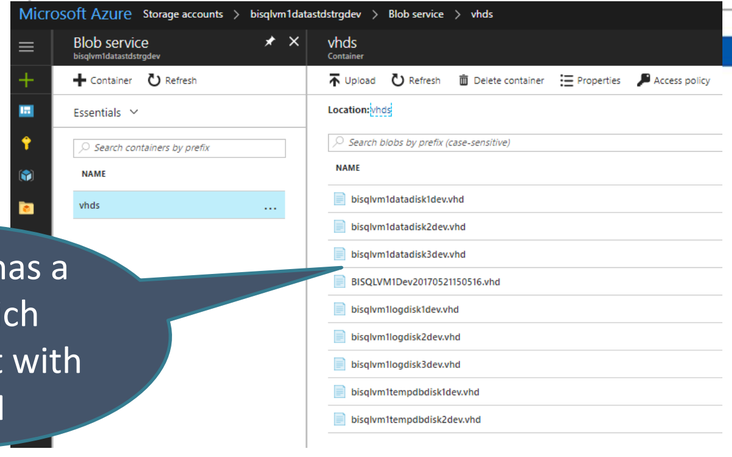
Increasing Performance of Standard Disks (2/3)



Before disk encryption has been configured



Note the single disk performance before it's collectively shared



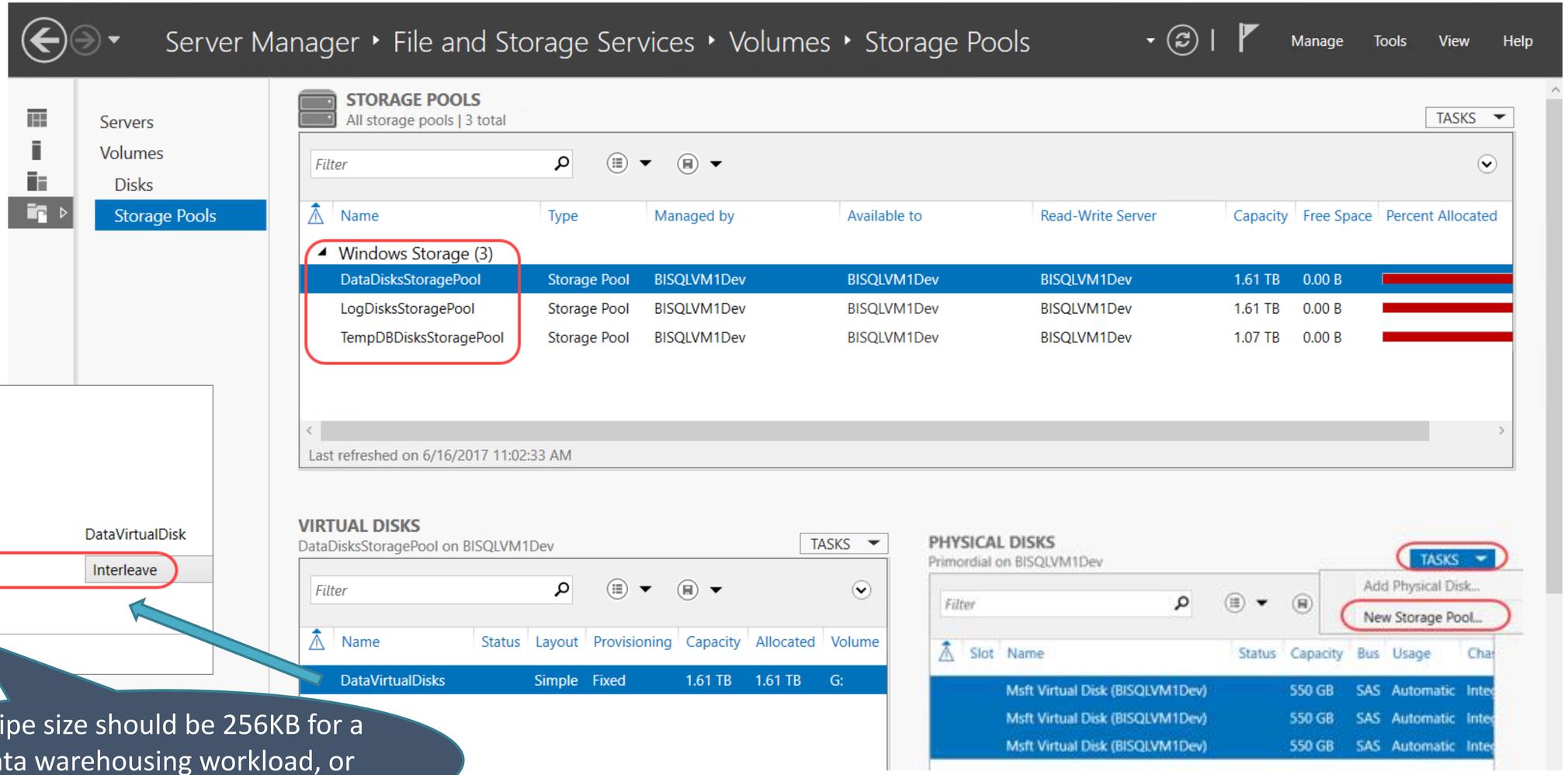
Each blob has a lease which associates it with the VM

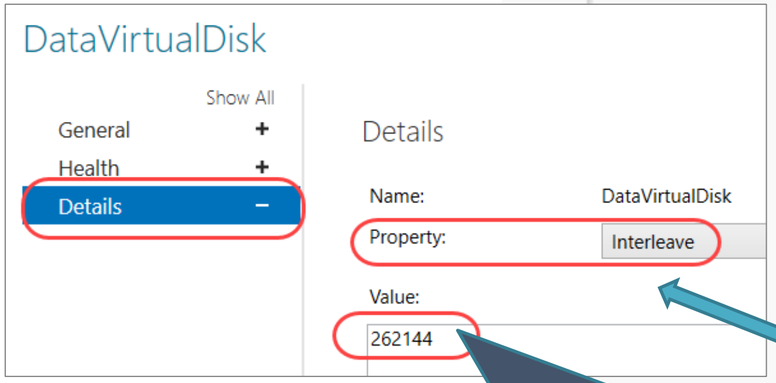
Disks attached to the VM

Blob storage container

Increasing Performance of Standard Disks (3/3)

This is a tradeoff: more complexity for greater performance at lower cost





Stripe size should be 256KB for a data warehousing workload, or 64KB for OLTP workload

Demo

Azure Storage Account

Tip #11

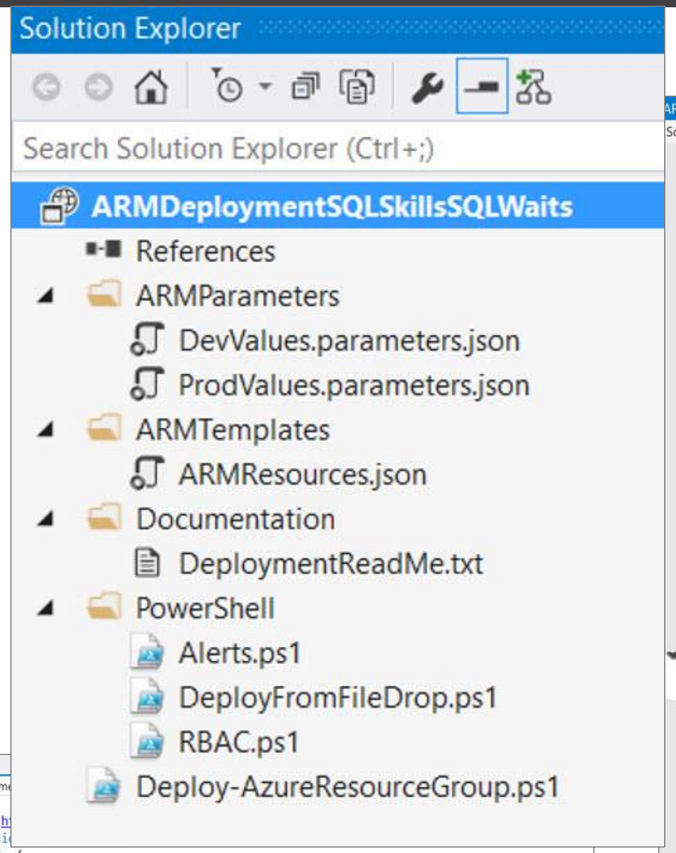
Know When to Leverage ARM and/or PowerShell

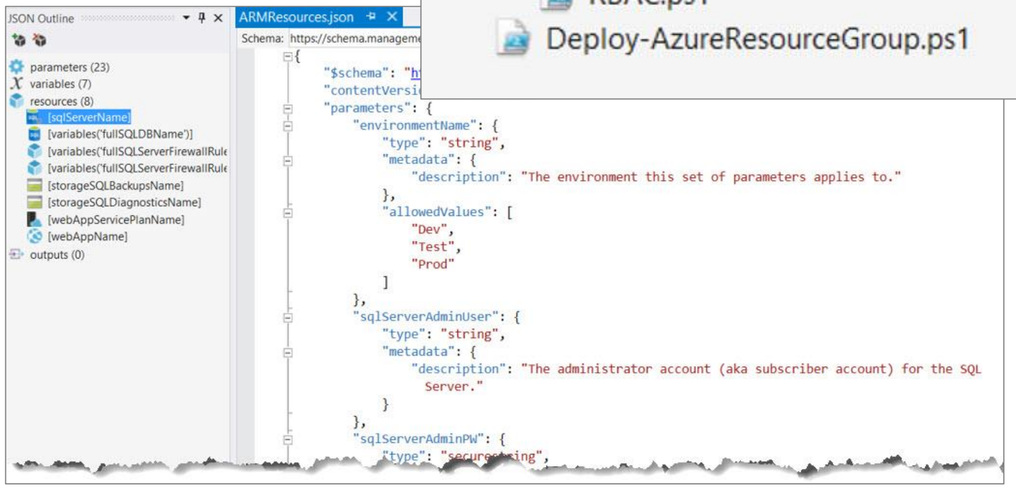
ARM Templates

Goal is for deployments to be modular & repeatable.

Invest time in learning how to do ARM templates so you can make a good decision when to use them.

This is known as "infrastructure as code" which can be checked into source control from a Visual Studio project.





Blog post on ARM deployments: https://blogs.sentryone.com/melissacoates/getting-started-with-automated-arm-deployments-in-azure/

ARM Templates

Use ARM if You Intend To…

- ✓ Include the configuration of Azure resources in source control (“Infrastructure as Code”), and/or
- ✓ Repeat the deployment process numerous times, and/or
- ✓ Automate deployments, and/or
- ✓ Employ continuous integration techniques, and/or
- ✓ Utilize DevOps principles and practices, and/or
- ✓ Repeatedly provision & de-provision testing infrastructure
- ✓ Do something unsupported by the portal interface

Starting from the automation scripts in the portal is a good way to learn, but you will want to customize & tidy them up

PowerShell

Automation is a balance of:
Wanting infrastructure in source code vs. little need for deployment repeatability

PowerShell scripts can be useful for:

- Deploying ARM templates
- Tasks unsupported in the portal
- Deploying specific items
  - Azure Data Factory
  - AAD service principal
  - etc…
- Managing resources
  - Assigning tags
  - Enabling disk encryption
  - Provisioning key vault
  - Defining alerts
  - Start/stop a virtual machine

```
#Input Area
$subscriptionName = 'InsertSubscriptionName'
$resourceGroupName = 'SQLSkillsWaitsLibraryRGDev'
$deploymentName = 'SQLSkillsSQLWaitsDeployment'
$templateFilePath = 'O:\Deployments\SQLWaits\2017-06-07\ARMResources.json'
$parameterFilePath = 'O:\Deployments\SQLWaits\2017-06-07\DevValues.parameters.json'

#ARM template deployment
New-AzureRmResourceGroupDeployment `
  -Name $deploymentName `
  -ResourceGroupName $resourceGroupName `
  -TemplateFile $templateFilePath `
  -TemplateParameterFile $parameterFilePath
```

From:
https://blogs.sentryone.com/melissacoates/getting-started-with-automated-arm-deployments-in-azure/

Recommendations for ARM and PowerShell

Use ARM for Provisioning Azure Resources When…

- Recognition of inter-dependencies is important
- Speed of deployment is important - ARM is faster than PowerShell or CLI (due to parallel deployment)

Use PowerShell For…

- Execution of ARM template
- Execution of misc. or piecemeal deployments
- Management of resources (ex: start/stop a VM) often in conjunction with Azure Automation
- Execution of action, resource deployment, or configuration not supported by ARM

Demo

Automation Script in Azure Portal

+

Visual Studio ARM Project

Key Takeaways

The pace of change in Azure is frequent—plan to keep up with announcements. New features can change a previous decision really fast. New services and features are constant.

Look at using PaaS unless you really need an IaaS solution – i.e., the simplest solution that gets the job done. And expect complications, delays, and a learning curve along the way.

Don’t shortcut the initial planning phase – do a POC if you can to learn. Don’t assume a feature is definitely available or works as you expect it to. For unfamiliar features or services, consider creating a sandbox area to test & learn on first.

Plan to invest some time learning ARM templates & PowerShell, if you haven’t already.

Pay attention to opportunities for cost savings. Developers & administrators have a huge impact on cost in the cloud.

Resources

Conceptual documentation:	https://azure.microsoft.com/
Reference documentation:	http://msdn.microsoft.com/
Documentation:	https://docs.microsoft.com/en-us/azure/
Samples:	http://github.com/azure-samples
Templates:	https://github.com/Azure/azure-quickstart-templates
Azure Architecture Center:	https://docs.microsoft.com/en-us/azure/architecture/