

February 6th-7th

nic 20/20 VISION

Oslo Spektrum



Azure SQL Database



What, why, why not?

Rune Ovlien Rakeie
Cloud Architect @ TietoEVRY

About me



Rune Ovlien Rakeie

Cloud Architect @ TietoEVRY in Norway

20+ years experience with database technology and SQL Server specifically, as Developer, DBA & Solution Architect.

Organizer of SQLSaturday Oslo

 @runeo34

@ rune.rakeie@evry.com



WHAT

On-premises/IaaS

Boxed

- On-premises - Physical/Virtual
- IaaS VMs /Azure, Google, AWS
- RDS for SQL Server (AWS)
- SQL RP on Azure Stack

Edge computing

SQL Database Edge

WHY

Microsoft
SQL Server

WHY NOT

PaaS

Synapse
Analytics

SQL Database
(PaaS)

Managed
Instance

Single
database

Elastic
Pool

nic

WHAT

WHY

WHY NOT

Design principles

1. HW/SW failure is inevitable
2. Operational staff make mistakes that lead to failures



WHAT

WHY

WHY NOT

Reasons for simplifying failure model

1. A fault-tolerant model requires you to deal with:
 1. Low-frequency failures
 2. Planned outages
 3. High-frequency failures
2. At cloud-scale:
 1. Low-frequency failures happens every week/day



WHAT

WHY

WHY NOT

Assumption

All components are likely to fail!



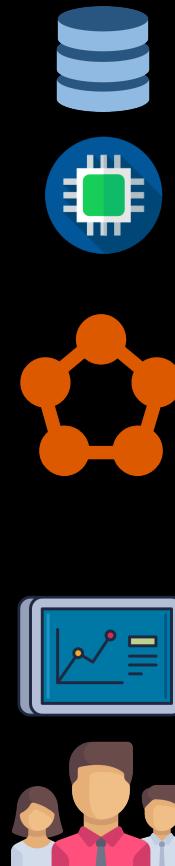
nic

WHAT

WHY

WHY NOT

Azure SQL DB (current) deployment status



| | |
|------------------------------|-------------------|
| Production databases | ~5.7 million |
| Number of cores | Millions |
| Number of nodes | 100s of thousands |
| # of Service Fabric clusters | Thousands |
| Typical cluster size | 5 – 250 nodes |
| New clusters/month | Hundreds |
| Regions | 56 |
| Daily telemetry volume | 10+ PB |
| Database Admins | 0 |
| Software Engineers (on-call) | ~500 |

nic

WHAT

WHY

WHY NOT

Azure SQL Database is:

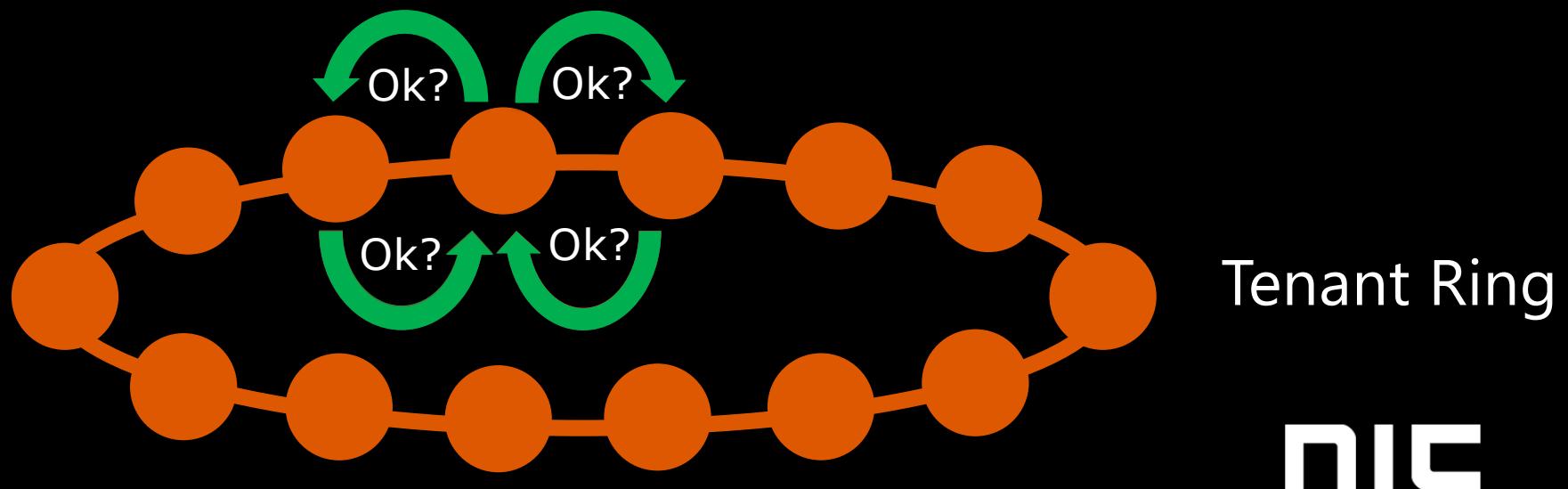
- Self-governing
- Self-healing
- A massively scalable PaaS
- Designed for 99.999%



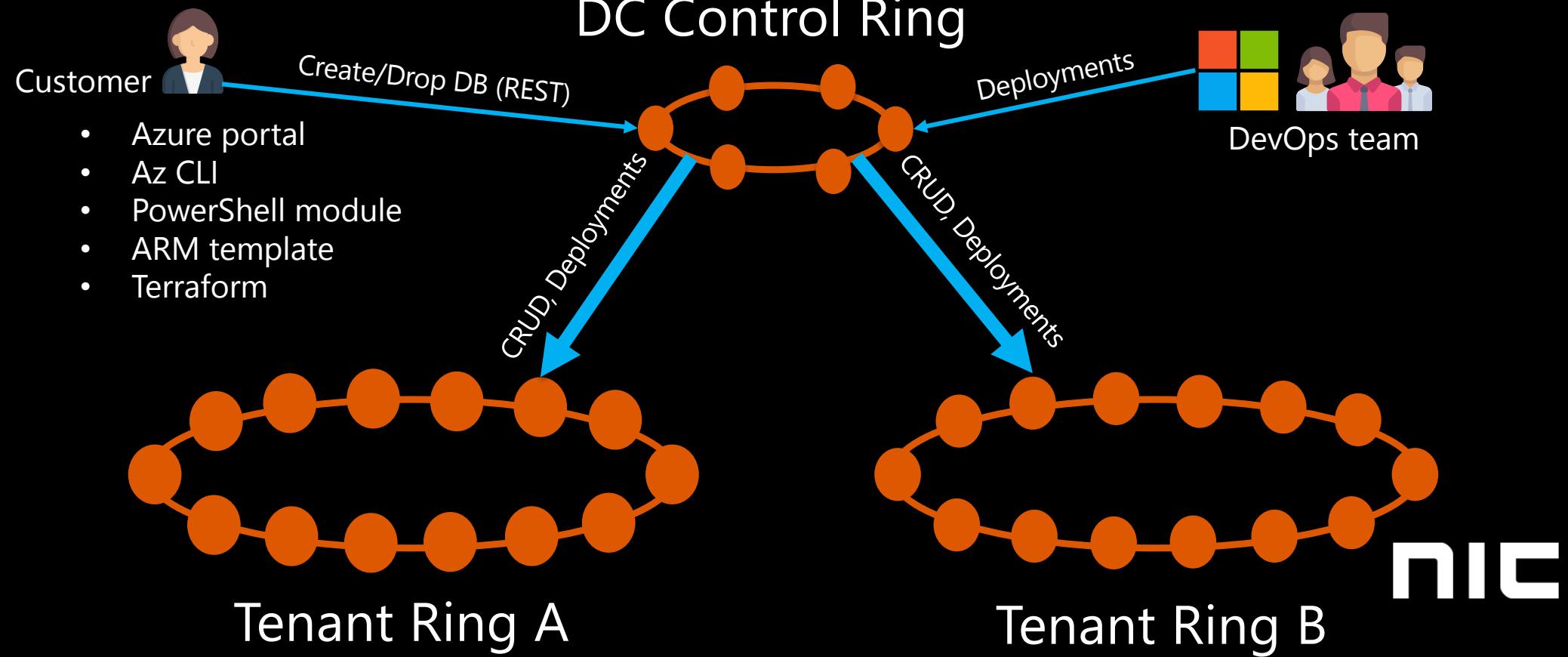
Azure SQL DB internals

Azure Service Fabric

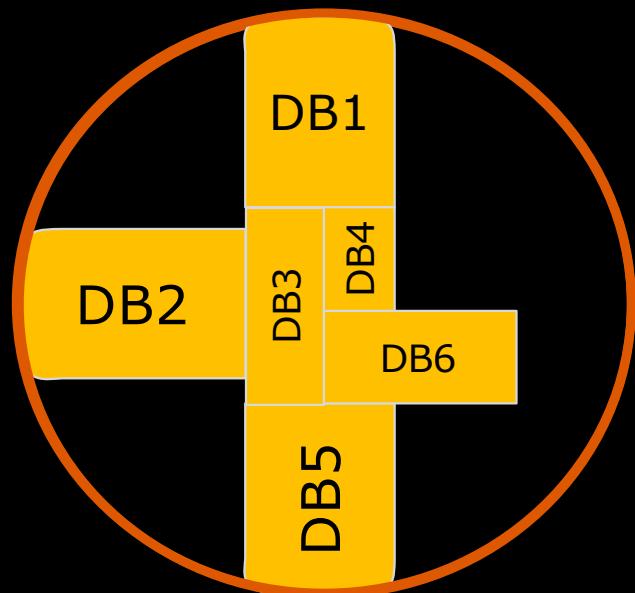
- Azure's version of Windows Failover Cluster



Azure SQL DB internals



Azure SQL DB internals



- One VM pr. node
- Fully utilizes underlying hardware
- Allocate compute/memory slice
- Throttling using Resource Governor

WHAT

WHY

WHY NOT

Performance tiers

1. General Purpose (Basic, Standard)
 - Most generic workloads/lighter OLTP
2. Business Critical (Premium)
 - Heavy OLTP, low latency
3. Hyperscale
 - VLDB, HTAP

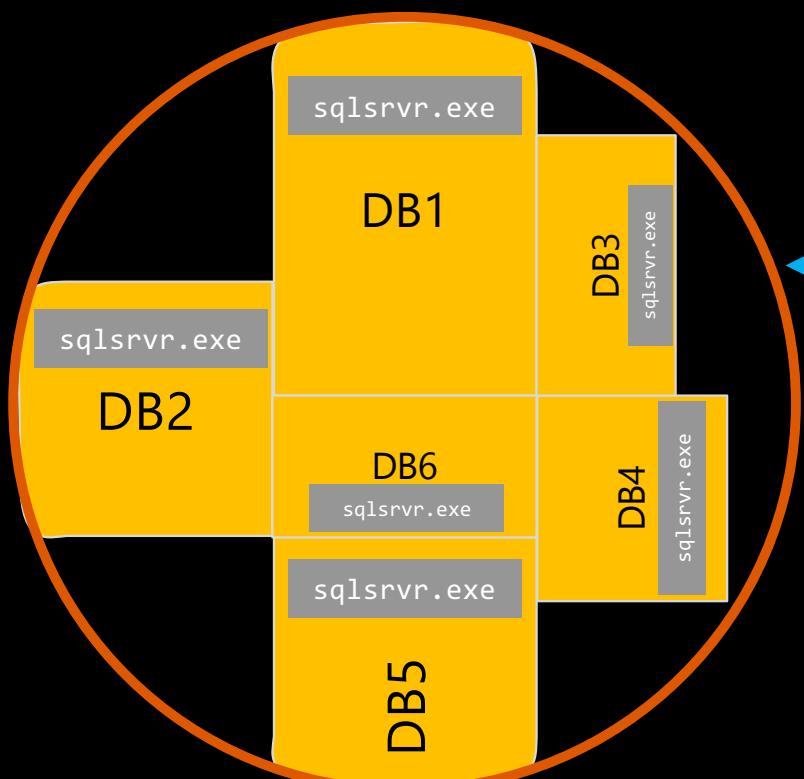
nic

WHAT

WHY

WHY NOT

General purpose



- Stateless binary
- If machine fails, database reconfigured within 30 seconds

nic

WHAT

WHY

WHY NOT



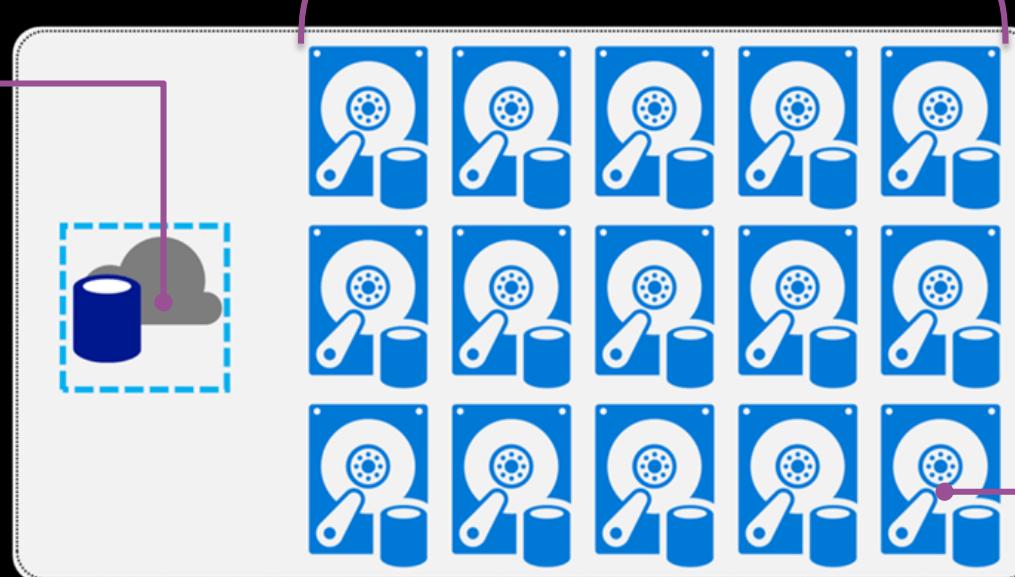
Managed
Instance

General purpose

Storage tuning

Azure Premium SSD disks

Stateless compute node



nic

WHAT

WHY

WHY NOT



Managed
Instance

General purpose

Storage tuning

Total sum of Azure Premium Disk sizes can't exceed 35 TB

| Premium SSD type | P4 | P6 | P10 | P15 | P20 | P30 | P40 | P50 |
|---------------------|------------|------------|-------------|-------------|-------------|----------------|----------------|----------------|
| Max disk size | 32 GB | 64 GB | 128 GB | 256 GB | 512 GB | 1024 GB (1 TB) | 2048 GB (2 TB) | 4095 GB (4 TB) |
| IOPS per disk | 120 | 240 | 500 | 1100 | 2300 | 5000 | 7500 | 7500 |
| Throughput pr. disk | 25 MB pr/s | 50 MB pr/s | 100 MB pr/s | 125 MB pr/s | 150 MB pr/s | 200 MB pr/s | 250 MB pr/s | 250 MB pr/s |

5 GB File = P10 Disk

300 GB File = P20 Disk

1.5 TB = P40 Disk

File size determines cap on throughput and IOPS

nic



Single
database

Serverless

General purpose (vCore)

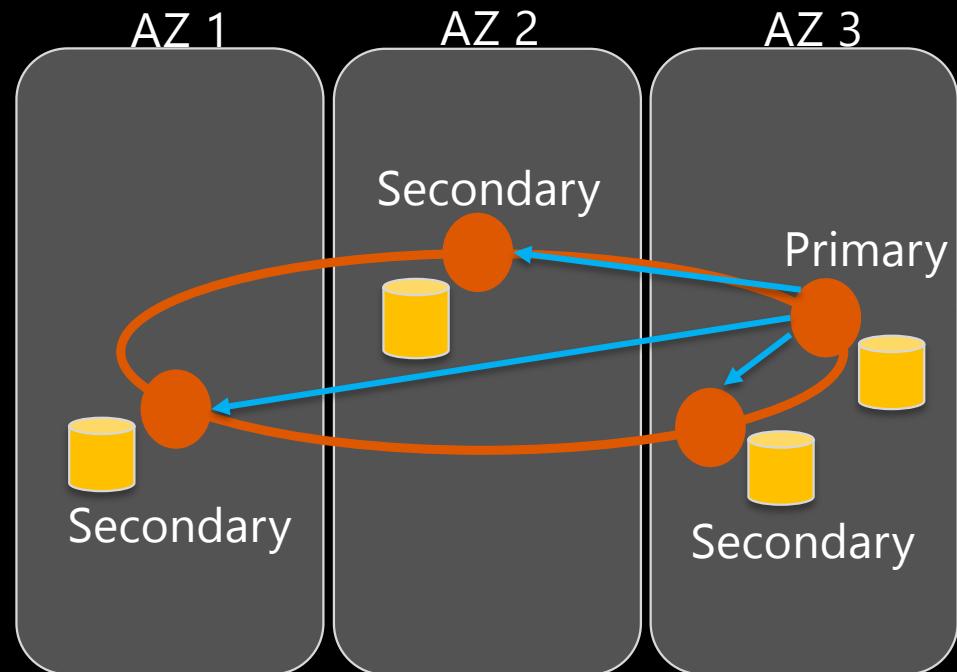
- Not really serverless...
 - Auto-pauses when not in use, and auto-resume when it is
 - Auto-scales based on workload
 - Billing is pr. **second** based on vCore and memory used
 - Only storage is billed when paused
- Best fit for:
 - Intermittent, unpredictable usage + lower average cpu use
 - Frequently rescaled databases
 - New databases without usage history

WHAT

WHY

WHY NOT

Business Critical



- Always On Availability Groups
 - Quorum commits – requires commit on primary and 2 secondaries
- Local SSD for storage
- Backup in Azure Storage
- Read-scale out
- Availability zones (preview)

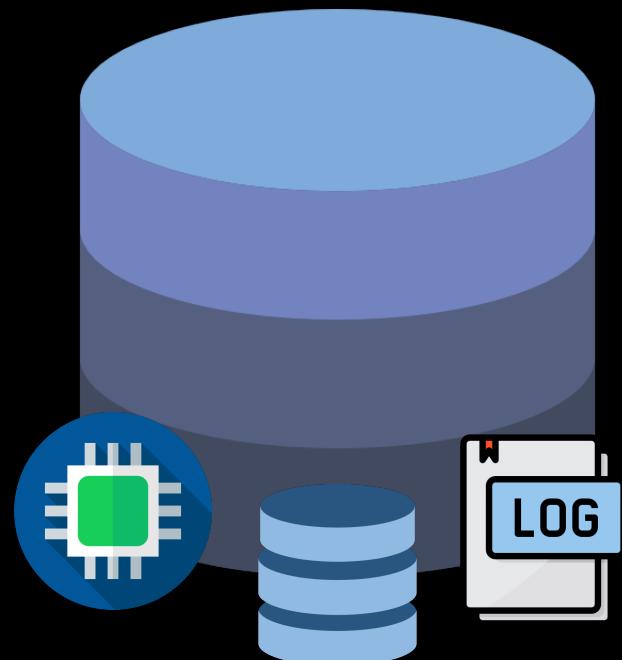


WHAT

WHY

WHY NOT

Hyperscale



Traditional database

nic

WHAT

WHY

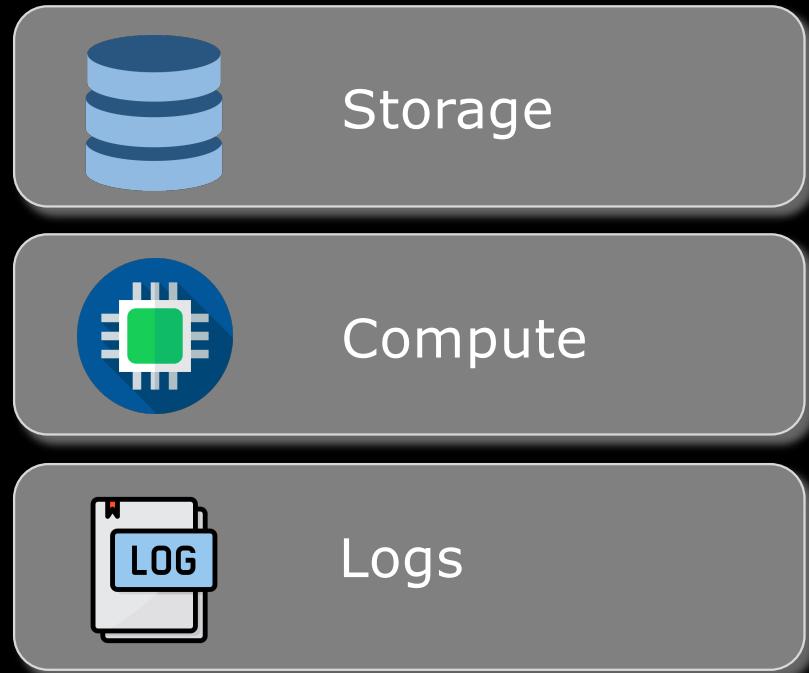
WHY NOT

Hyperscale



Traditional database

Decoupling



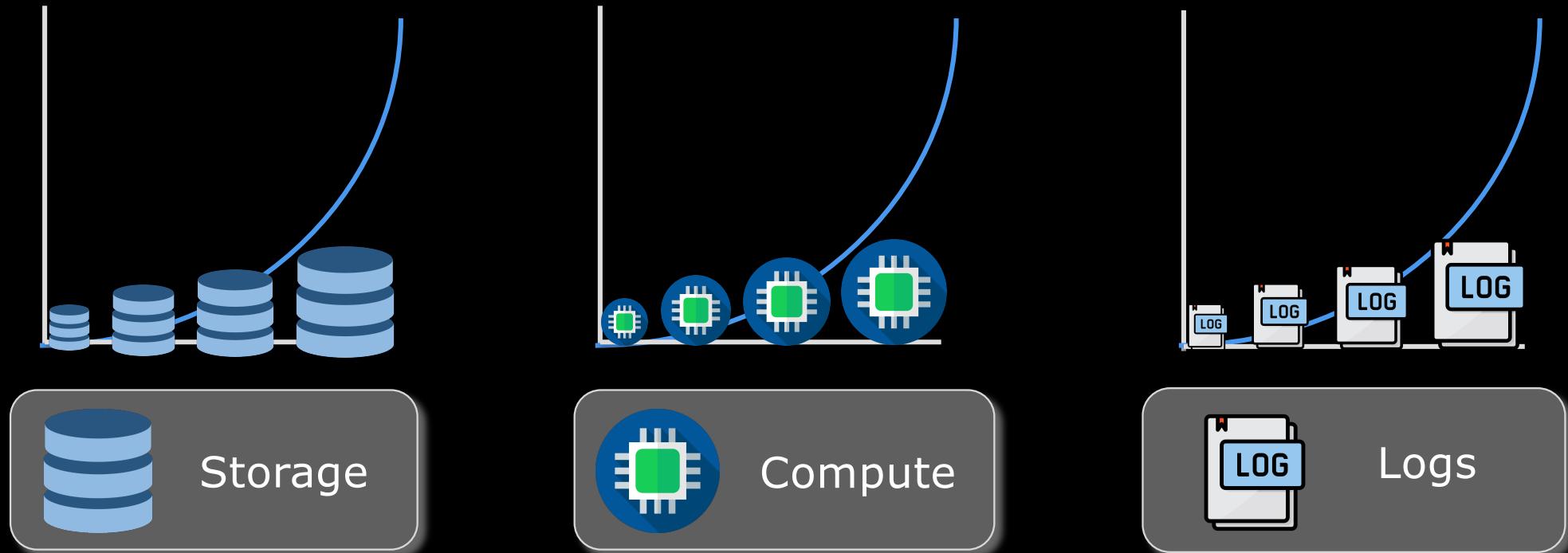
nic

WHAT

WHY

WHY NOT

Hyperscale



Scales independently

nic

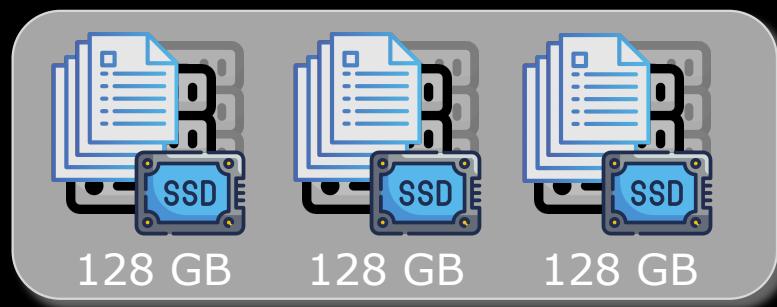
WHAT

WHY

WHY NOT

Hyperscale

Page servers



- Microservice based architecture
- Secondaries for resilience
- Local SSD cache pr page server
- Scales horizontally up to 100TB

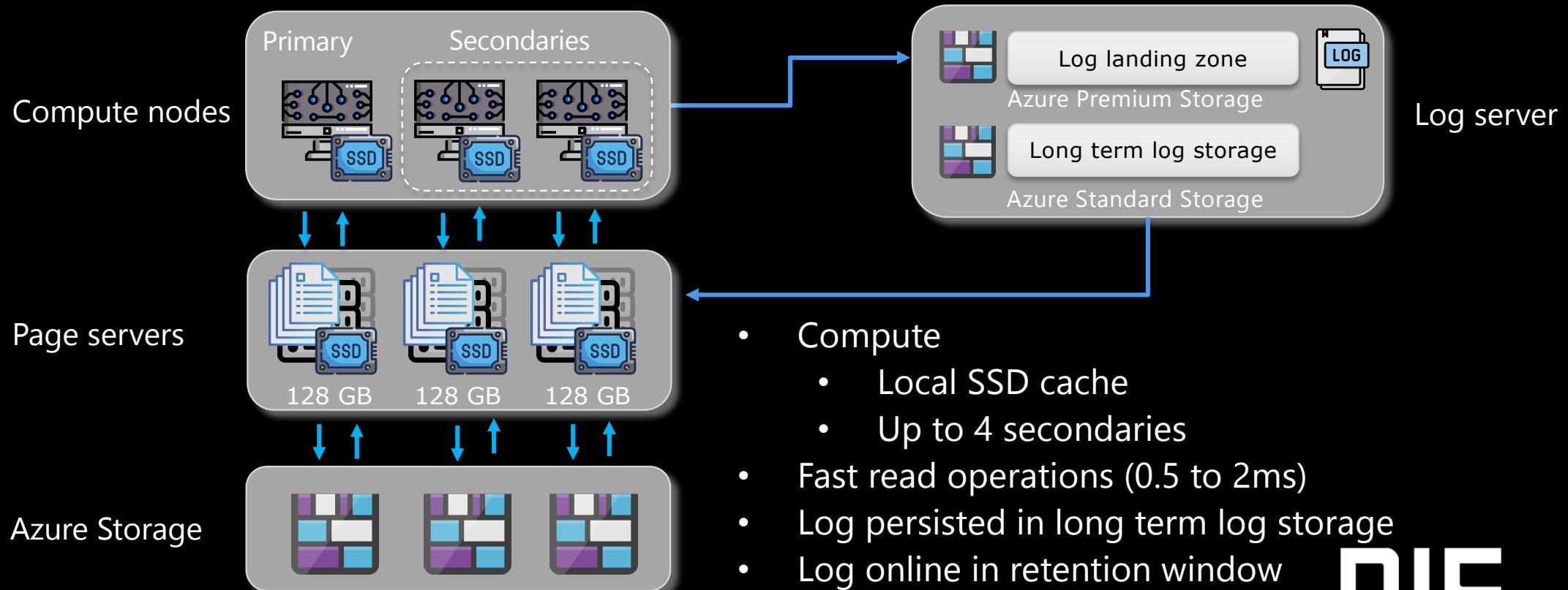
nic

WHAT

WHY

WHY NOT

Hyperscale

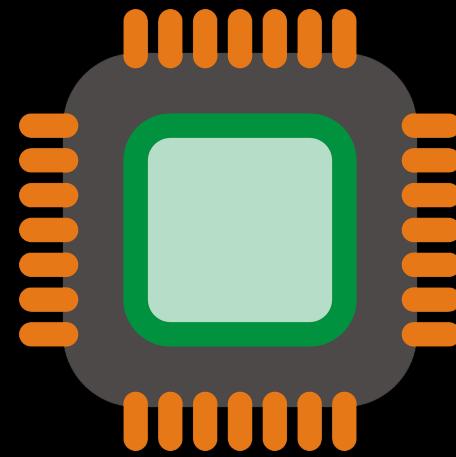


WHAT

WHY

WHY NOT

Purchasing models



DTU

vCore

nic

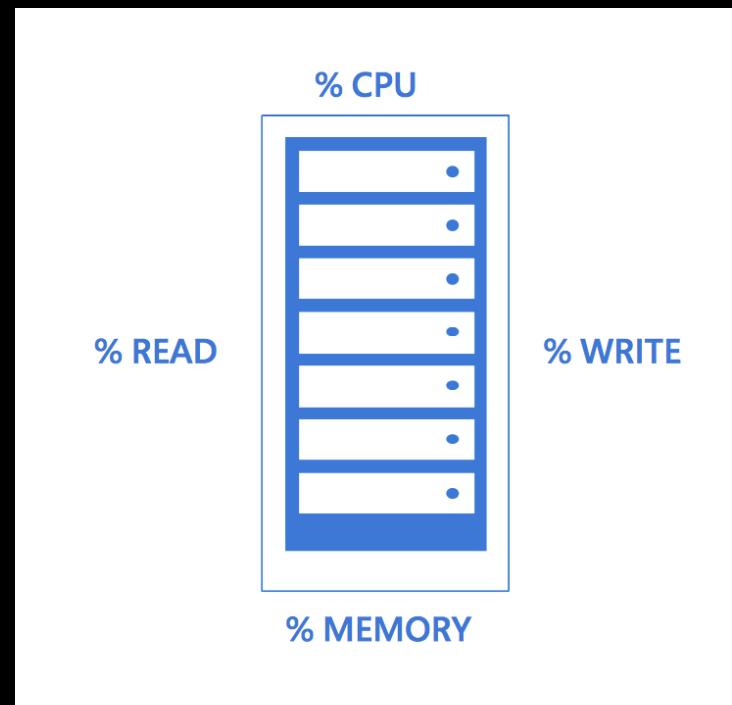
WHAT

WHY

WHY NOT

What is a Database Transaction Unit?

- The relative power assigned
- Blended measure



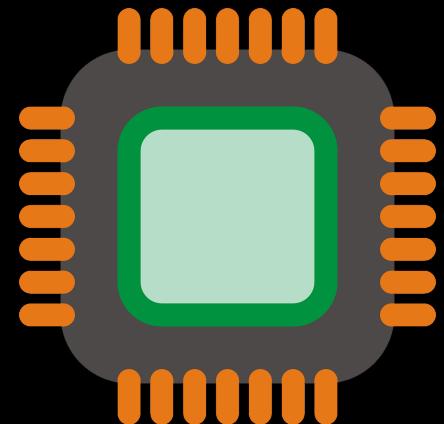
WHAT

WHY

WHY NOT

vCore

- Transparency in the hardware details
- Supports Azure Hybrid Benefit and Reserved Instance



| | <u>Gen 4 (Haswell)</u> | <u>Gen 5 (Broadwell)</u> | <u>Fsv2-series (only GP)</u> | <u>M-series (only BC)</u> |
|-------------|------------------------|--------------------------|------------------------------|---------------------------|
| Clock speed | 2.4 GHz | 2.3 GHz | 2.7 GHz | 2.5 GHz |
| Max cores | 24 | 80 | 72 | 128 |
| Core type | Physical | Hyperthreaded | Hyperthreaded | Hyperthreaded |
| Memory | 7 GB pr core | 5.1 GB pr core | 1.9 GB pr core | 29 GB pr core |

nic

<https://docs.microsoft.com/en-us/azure/sql-database/sql-database-service-tiers-vcore>

WHAT

WHY

WHY NOT



Managed
Instance

Azure SQL Managed Instance

SQL Server compatibility

- Fully-fledged SQL instance with nearly 100% compatibility with on-premises

Fully managed DBaaS

- Built on the same infrastructure as SQL DB
- All DBaaS features

Full isolation and security

- Contained within your VNet
- Private IP addresses
- Express Route / VPN connectivity

nic

WHAT



Managed Instance

What about BI services?

Not installed side-by-side with Managed Instance

Migrate your SSIS packages to new SSIS on Azure Data Factory (PaaS service)

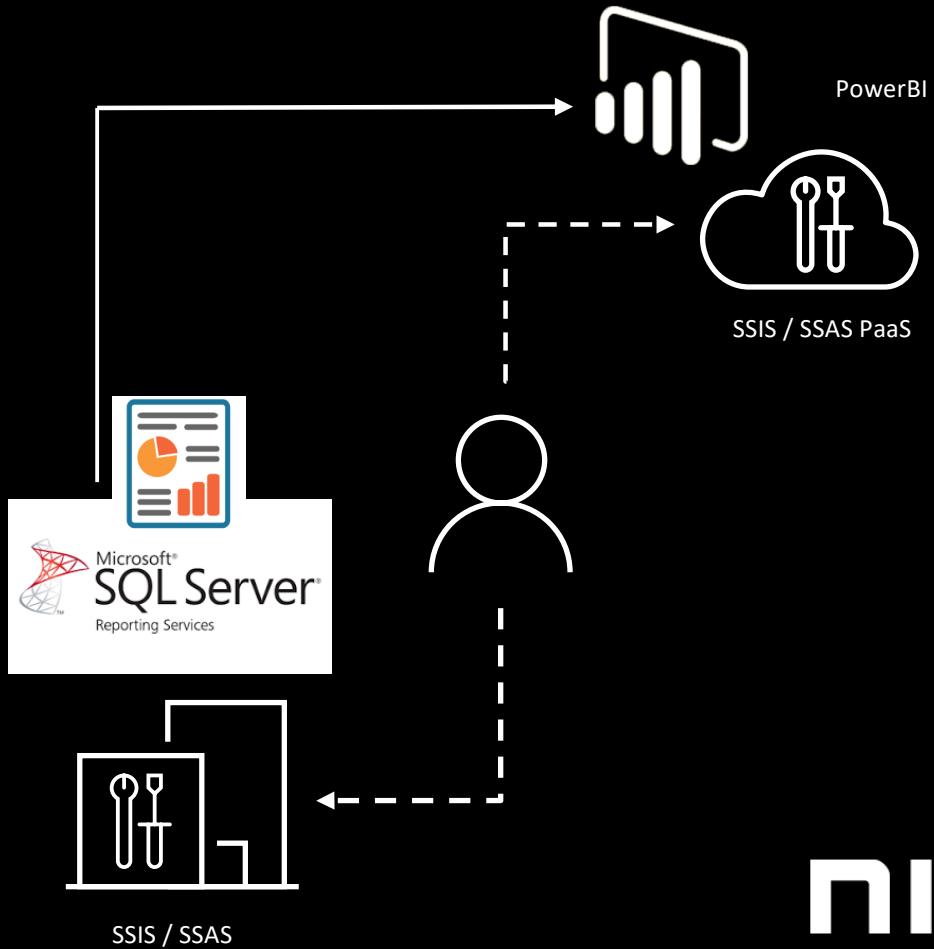
Migrate your OLAP models to Azure Analysis Services

... or run these services in Azure virtual machines

For SSRS: run in a virtual machine, or switch to Power BI

WHY

Recommendation - move BI solutions to PaaS model



nic

WHAT

WHY

WHY NOT



Single
database

Azure SQL Database Singleton

Purpose

- Optimized to reduce overall management cost

Fully managed DBaaS

- Upgrades
- Patches
- Auto perf. Tuning
- Threat Protection

Business continuity

- Built-in fault tolerance
- Automated backups
- Point-In-Time Restore
- Geo-restore
- Geo-replication

nic

WHAT

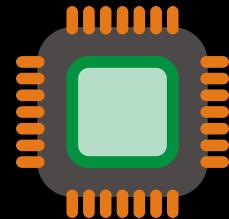
WHY

WHY NOT



Single
database

Azure SQL Database Serverless



Purpose

- Price-performance optimized
- Saving cost

Capabilities

- Auto-scaling
- Auto-pause
- Auto-resume

Use cases

- Intermittent or unpredictable usage pattern
- Frequently rescaled
- Difficult or not possible to estimate before deploy

nic

WHAT

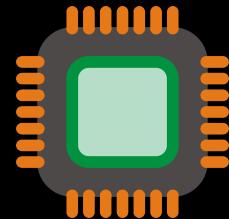
WHY

WHY NOT



Single
database

Azure SQL Database Hyperscale



Purpose

- Highly scalable storage
- Compute performance tier
- Optimized to reduce overall management cost

Capabilities

- Support for 100 TB
- Snapshot backup
- Fast database restores
- Higher log throughput
- Rapid scale out/up

Use cases

- Have very large databases
- When you are limited by 4 TB
- Seek high performance/scalability
- Optimized for OLTP.
- Supports hybrid and analytical processing workload

nic

WHAT

WHY

WHY NOT

Comparison – Azure SQL Database

| | <u>Managed Instance</u> | <u>Single Database</u> | <u>Elastic Pools</u> |
|--------------------|-------------------------|--|--|
| Scheduled Activity | SQL Agent | Elastic Database Jobs | Elastic Database Jobs |
| Cross DB queries | Native Capability | Elastic Query | Elastic Query |
| Feature Scope | Instance | Database | Database |
| Compute measure | vCore | DTU/vCore | DTU/vCore |
| Database Limit | 100 | Single Database | Varies by tier |
| vNet Connectivity | Native | Via Service Endpoint Via Private Link (preview) | Via Service Endpoint Via Private Link (preview) |



WHAT

WHY

WHY NOT

Comparison – Managed Instance vs IaaS

| | <u>Managed Instance</u> | <u>Retail SQL Server</u> | <u>RDS for SQL Server</u> |
|--------------------|-------------------------|--------------------------|---------------------------|
| Scheduled Activity | SQL Agent | SQL Agent | SQL Agent |
| Cross DB queries | Native Capability | Native Capability | Native Capability |
| Feature Scope | Instance | Instance | Instance |
| Compute measure | vCore | Core Count | Core Count |
| Database Limit | 100 | 32767 | 30 |
| vNet Connectivity | Native | IaaS VM/Hybrid VPN | Native to AWS |



WHAT

WHY

WHY NOT



Single
database



Managed
Instance

1. You don't have a database administrator on staff

- No hardware to maintain
- OS always updated and patched
- DBMS software always updated and patched
- Automatic backup, <= 35 days PITR and LTR <= 10 yrs
- Built-in high resilience and redundancy
- DR through geo-replication (one-click)
- 99.99% SLA

nic

WHAT

WHY

WHY NOT



Single
database



Managed
Instance

2. You're concerned about security

- Encrypt data at rest (TDE), in transit (SSL/TLS)
- Always Encrypted in secure enclaves
- Advanced Threat Protection
- Discover, track and remediate vulnerabilities

nic

WHAT

WHY

WHY NOT



Single
database



Managed
Instance

3. Your CFO hates CAPEX, but ❤️ ~~OPEX~~

just a chance
to save money

- ✓ Pay-as-you-go model
- ✓ Highly elastic
- ✓ Reserve capacity – save up to 33%
- ✓ Reuse existing licenses – save up to 55%
- ✓ Combine the two for savings for up to 80%
- ✓ Utilize Serverless deployment for savings

nic

WHAT

WHY

WHY NOT



Managed
Instance

4. Dependencies not compatible with Singleton

- “Lift-n-shift”-scenario. Support >= SQL Server 2005
- Need SQL Agent
- Need support for one or more of :

| | |
|------------------------|---------------------------|
| Cross-database queries | Transactional Replication |
| Linked Servers | Change Data Capture |
| DB Mail | CLR (limited) |
| Service Broker | Global temp tables |



Managed Instance

When to choose

Vendor Apps

SQL Server back end

- DB Compatibility

Needs to support SQL Server Authentication or AAD Authentication

Where databases have multiple files and filegroups

In-House Apps

Can't easily be re-factored to use Azure SQL Database

Refactor to use AAD or fall-back to SQL Authentication

Control needed for files and filegroups

Need SQL Server Agent

New Cloud Apps

Where older design patterns are used

Cross database queries that can't use Elastic Query

- Linked Servers etc.

Need to leverage T-SQL surface area that is not available in SQL DB



WHAT

WHY

WHY NOT



Single
database



Managed
Instance

1. You have one or more showstoppers

- App only support Windows AD Authentication
- App use unsupported DDL/DML statements
- You need MSDTC
- Have more than 100 databases (MI)
- Database larger than 4TB (8TB for MI on GP)
- Your app vendor says they don't support it

WHAT

WHY

WHY NOT



When NOT to choose

| <u>Vendor Apps</u> | <u>In-House Apps</u> | <u>New Cloud Apps</u> |
|---|--|--|
| Windows AD Auth Only > 100 databases on an existing SQL Server Single file > 4 TB | Filetable/filestream functionality > 100 databases on an existing SQL Server Uses the In-Memory Engine | Really small databases(s) with low resource use < 10 databases involved New Cloud Apps should be built on Azure SQL DB!! |



Azure SQL Database



What, why, why not?

Rune Ovlien Rakeie
Cloud Architect @ TietoEVRY

February 6th-7th

nic 20/20 VISION

Oslo Spektrum

