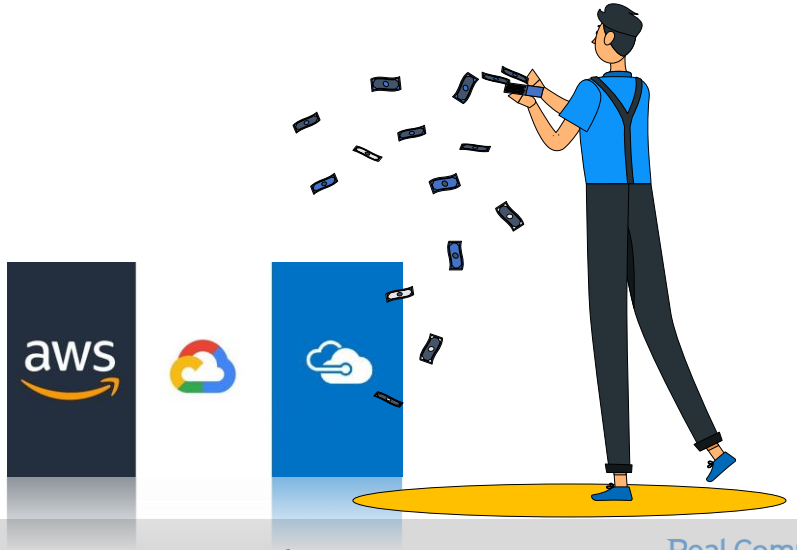
#datasatpn

Data Saturday #1

February 27th, 2021

How to use “the Cloud “ for Data

…and actually save money



altitudo [α SYS] beanTech IT moves your business CARGO BI Insurance Data Analytics fluentis QUANTUMDATIS Real Comm Easy for real

“If someone asks me what cloud computing is, I try not to get bogged down with definitions. I tell them that, simply put, cloud computing is a better way to run your business

—Marc Benioff, Founder, SalesForce

About me

Emanuele Meazzo

DBA, BI Facilitator, Azure Data Engineer, gets anxious when something is not optimized

Contact Info:

- Email: emanuele.meazzo@gmail.com
- Blog: tsql.tech
- Linkedin: /emanuelemeazzo
- GitHub: github.com/EmanueleMeazzo/tsql.tech-Code-snippets

What's this about

- This is a not deeply technical presentation
  But it's cool since you can bring some of these points to your boss
- Hopefully, everything should be trivial for you
- If you're already "in the cloud" and any of this isn't trivial,
  sit down, we have to talk
- The argument is so vast we could talk for hours,
  but there's only 60 minutes so there will be gaps,
  we can continue at a different time, just ping me
- Unfortunately, you're going to leave this session with
  more questions than answers
- Slides are text rich in order to be of reference for the future, you don't need to read it all

The State of the Cloud

82%	Of enterprises indicate that managing the cloud spend is a challenge
77%	Of organizations use cost efficiency and savings to measure cloud progress
73%	Plan to optimize the existing use of cloud (cost savings), making it the top initiative for the fourth year in a row
50%	More than 50 percent of enterprise workloads and data are expected to be in a public cloud within 12 months
30%	It's estimated that organizations waste 30 percent of cloud spend, are over budget by an average of 23 percent and expect cloud spend to increase by 47 percent next year

Source: Flexera State of the Cloud Report 2020

CapEx vs OpEX

AKA Why the business wants to move to cloud

CapEX

Capital Expenses

Consist of the funds that companies use to purchase major physical goods or services that the company will use for more than one year

Usually approved through several layers of management (including executives), which will hold up purchasing until approval is received.

OpEX

Operational Expenses

Represent the day-to-day expenses necessary to run a business.

OpEx are fully tax-deductible in the year they are made.
This can be an attractive accounting option if the company has limited cash flow and wants to be able to deduct the total cost of an item in a tax year.

New companies and the Cloud

It’s usually a no-brainer to start cloud-first

Cloud as a startup advantage

No upfront investment in architecture
Lower capital risks
Faster creation of a Minimum Viable Product
Faster business pivoting

Cloud as a business model

A SaaS product is easier to try and to adopt
Saas Profit = EBITDA

Existing companies and the Cloud

This is where the real planning and careful evaluation is needed
The following questions must be asked first:

1

What is my goal?
Cost savings?
New Product?
Faster Performances?

2

What does my current infrastructure look like?
Is the hardware at EOL?
Or is it just slow?

3

What technologies my team knows deeply?
Am I willing to invest in their formation and/or in new hires?

Speed

~50

Azure updates per Quarter

The Cloud is FAST

https://azure.microsoft.com/en-us/updates/

Types of Cloud Services

IaaS
Infrastructure as a Service

Getting Rid
Hard

PaaS
Platform as a Service

Getting rid of anything but the software configuration

The DataLake trap

I've seen countless companies burning large amounts of cash in search for the "big data" solution, only because the management was driven by the Big Data hype

85%

Is the reported failure rate of Big Data enterprise projects

Avoid dumping data into a big bucket just because "now you can" using cloud tools.
Saving money means first and foremost not throwing them away in a project that will never see the light of the day.

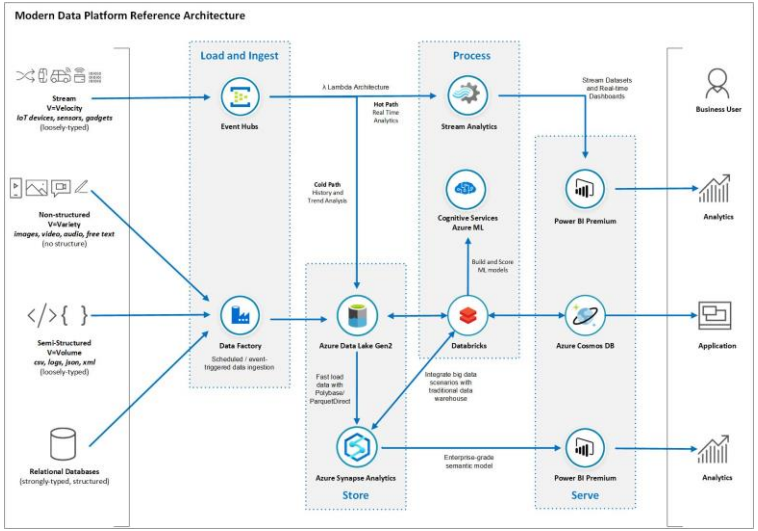
Ask yourself: What is the problem that I'm trying to solve with this?

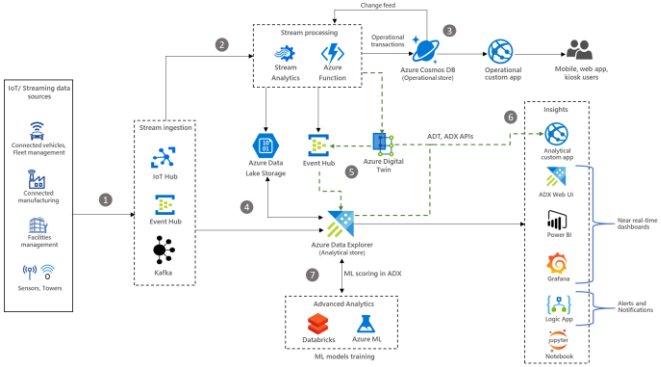
Source

The trap, continued

Choose the right technology for you.
The market is full of buzz-words, everything is real time, petabyte scale, streaming rich data.

The first question should be, what do I need to do?
What's the problem I'm trying to solve?
What's the business requirement behind all of this?





In the cloud architecture examples you see on the web, you'll see lots of services communicating with each other

Of course, any vendor would be very happy to sell you all the cloud services you need, or even better everything they think you need

Trivia

Things you can say to your cloud vendor but not your partner

I want it to be FAST and CHEAP

My apps need to change !?!?

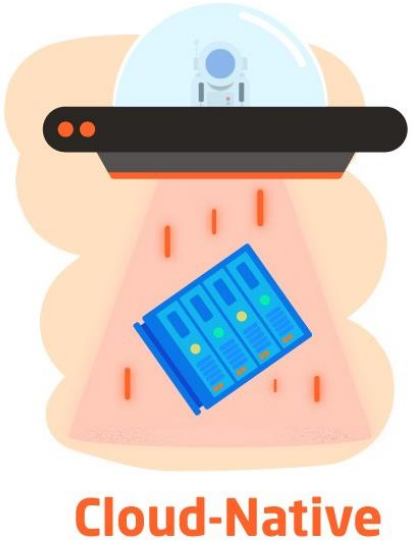
That's the main point of all of this, there is no real Cloud transformation without a change in your apps

No more "It always has been done like this"
If you want to do what you've always done, but in the cloud, you'll rarely see real benefits from it.

Let's see some of the typical hurdles of cloud adoption

Lift & Shift

Vs



- A classic business *move* to get to the cloud “fast”
- Can be a good short-term solution (or a long term one for non business critical systems)
- Infrastructure costs are higher than continuing to use your servers, if you already own them
- Cannot use 90% of the features that makes the cloud more efficient
- You’re basically renting someone’s else server instead of your own

Image Source

Multiple Vendors?

Everything in the same Cloud?

PROS

- (possibly) Lower TCO
  - Bundle licenses & discount tiers
  - No egress charges if in the same geographic area
  - Reduced data movement
  - Reduced ETL complexity
- Easier to focus on optimizing a specific platform
- Ability to hire specialized developers

CONS

- (possibly) Suboptimal feature set
- If your provider decides to deprecate a critical feature for your business, you're screwed
- In case of a cloud provider that's no longer competitive, the whole IT stack needs to be moved over

Use multiple Cloud Vendors?

PROS

- Lowered Risk (?)
- Higher Flexibility
- Ad-hoc solutions for specific requirements

CONS

- Higher Costs
  - Egress charges for systems to communicate between cloud providers or between cloud and on-premise
  - ETL costs to integrate data coming from one system to the other
    - Creating, Running, Maintaining, Evolving
  - Knowledge
- Ad hoc integration

[Should you put all your eggs in one basket? Thinking about Database eggs and Cloud baskets | T-SQL Tech]

Latency

Especially if you Lift&Shift parts of your application/data to the cloud, latency can bite you

Not necessarily because cloud latency is bad

Maybe you're used to the sub-ms latency of your server which was sitting in the closet behind you, and now your chatty application is trying to get data from the other side of the world.

For existing apps, verify the level of chattiness and how latency affects them

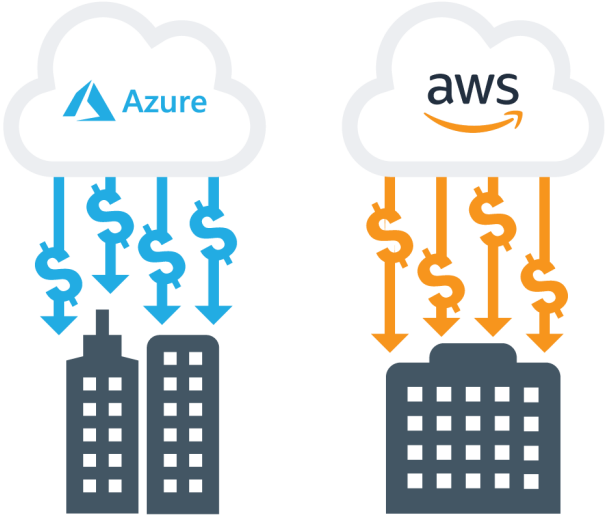
Egress Fees

When you start moving to the cloud, you'll have flows of data in and out of cloud resources
Be careful, it's not like your good 'ol datacenters, you're going to pay for it

While importing data is mostly free, moving data from the cloud to on prem always costs you money

Azure Sample Pricing:

- 5GB-10TB/Month: €0.0738 per GB ~ 755€/month (10TB)
- 10TB/50TB/Month: €0.0700 per GB ~ 3622€/month (50TB)



So, be careful when moving big amounts of data.
But wait, there's more!

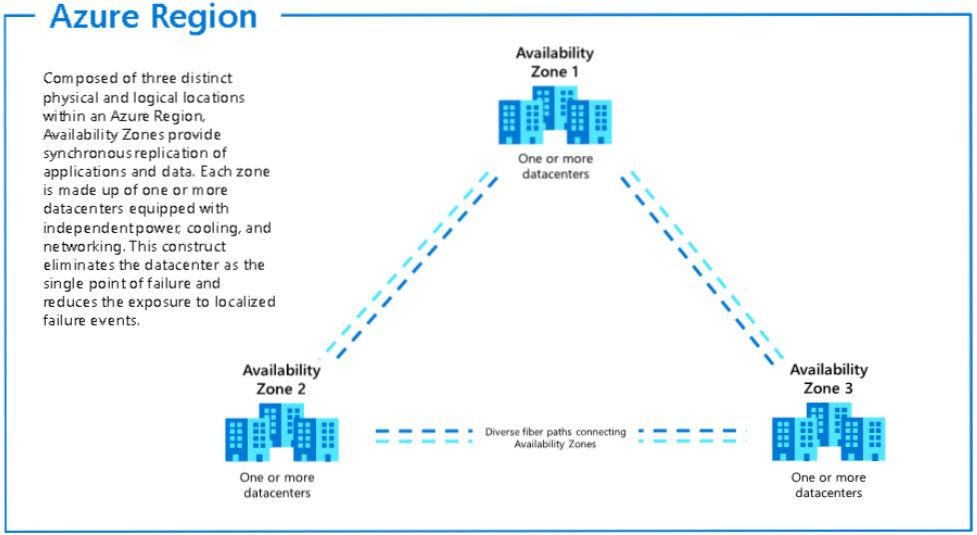
You pay for intra-cloud data movement too!
And I'm not talking like between different cloud providers

Inter-region data transfer is billed (albeit at a lower rate)
Finally, data transfer between Availability Zones is billed too (starting from July for Azure)

Control where your data is

After the two previous slides, you should have realized that data location is somewhat important
It's all a game of balancing the following:

- **Region exclusivity:** Keep all traffic within the same region.
  If traffic needs to exit a region check and choose the region with the lowest transfer rates
- **Region Vicinity:** The closer the cloud region to your premise and/or users, lower the latency
- **AZ exclusivity:** In both AWS and Azure traffic within the same Availability Zone is free.
- **Geo Replication:** Critical data, that needs a very tight SLA should be replicated, make sure it's worth it



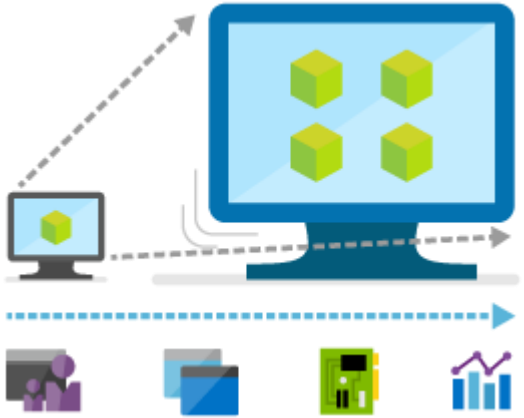
Scaling Services

You can get most of the cost savings by simply being able to scale up/down cloud resources, or even better being able to turn them off completely

If you're using a cloud VM like an on-prem one, in a never-changing configuration, what's the point of it being physically more distant from your premise and managed by a third party?

At the same time, the scaling considerations to make are Different than on-prem:

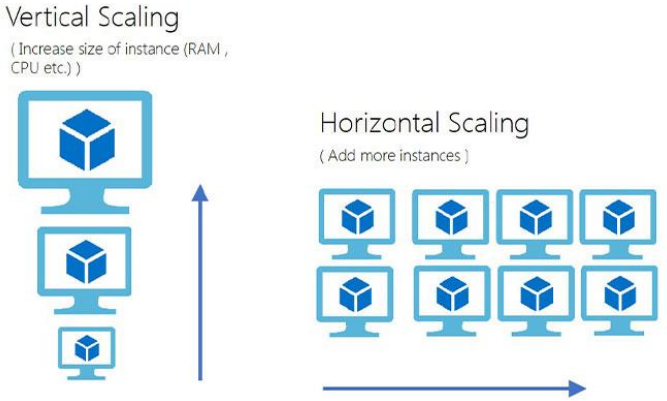
- How can this service scale?
- Can it be turned off?
- How long does scaling take?
- Does my application support horizontal data scaling or does it need to be baked into the data layer?
- Does this dev instance really need to stay on 24/7 ?



More on Scaling

Like any other solution, however, scaling introduces some problems to solve, for example:

- Depending on the service, scaling up and/or down may not be seamless, and service interruptions may happen (from a simple drop of in-flight connections to various minutes of service interruption)
- Due to the above, any cloud-ready application should integrate a retry logic for any call to any service, especially to the persistence layer
- Does the service scales automatically up and down depending on the load, or you have to programmatically scale it yourself?
- While vertical scaling can be almost transparent (except the downtime), a horizontal scaling isn't, if your application isn't already in the cloud mood; do you even know how it scales?
- If your data starts getting spread across various locations, you really have to think about how to distribute it



Performance is (literally) Money

Scaling up can happen for two reasons:

- A good one: an increased request volume to your system
- A bad one: a performance problem that's being fixed by blindly throwing hardware at it

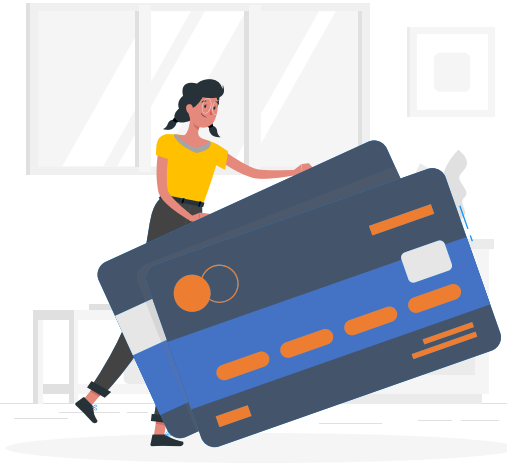
While the short-term solution for a performance problem could be just throwing more resources at it, it should be just a way to get more time for your devs/dba to fix the performance issues.

Since in the cloud we're basically renting someone's else hardware, using way more than necessary because no effort has been spent in optimizing the data access is an **expensive** mistake.

Performance Tuning on the cloud will save you money by the hour | T-SQL Tech

Reservations on Reservations

Reserving capacity on the cloud is a nice way to save money.
All the major Cloud vendors offer discounts when opting for reserving a service for 1+ years, even up to 60% on the overall TCO



This is of course a great way to save money, but only if you do the following:

- Make sure that the resource must stay online for a percentage of time greater than the discount applied
- Measure your current workload
- Make sure that you're buying a resource be appropriate for the whole reservation period
  (While you can often exchange your reservation for another resource, it's usually only for upselling, not down)
- Understand that you're reserving the price, not the hardware itself

Kill unused resources

Creating resource on the cloud is very easy, so much that's not uncommon to create something just for testing purposes and leaving it be for the time being.

The fact that some vendors makes it purposefully difficult to have an overview of everything that's deployed on a specific subscription surely helps with the above.

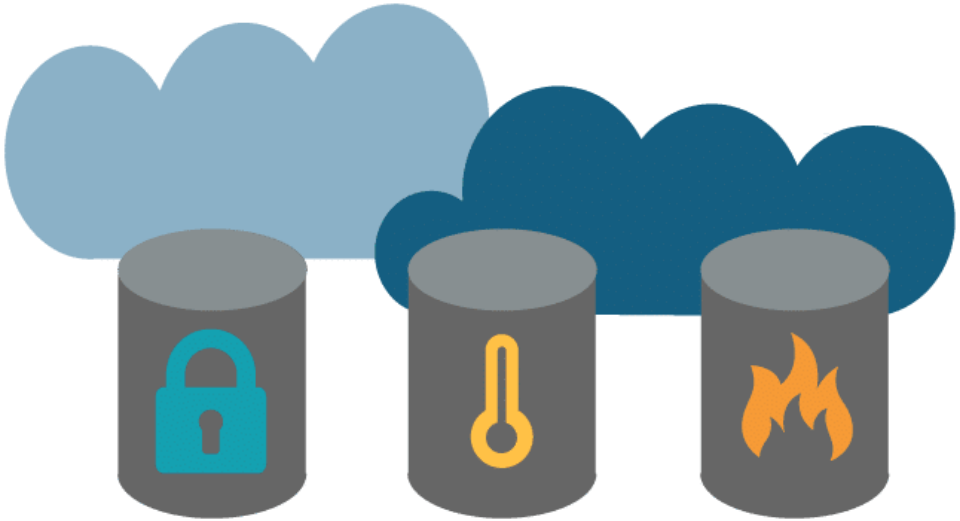
- Review your cloud spending every month
- Keep an eye on your resource inventory
- Use the integrated vendor tools for budgeting and cost allocation to identify which project/area is responsible for the most expenses and optimize it

Storage Tiering & cleaning

Storage is “cheap” but when you start storing all kind of data in the cloud the bill starts to grow too

- If you have data that you **really** know you’re not going to use, ever, get rid of it
- For all the rest, choose an appropriate storage tier:

  “Hot” data is accessed frequently and/or needs a fast retrieval, hence it needs a fast storage



“Cool” data doesn’t have the same performance requirements, so, it can be stored someplace where you can wait a little bit to fetch your data

“Archive” is the cloud equivalent of tapes, where you can wait for hours to get your data, but it’s cheap

The Cloud DBA

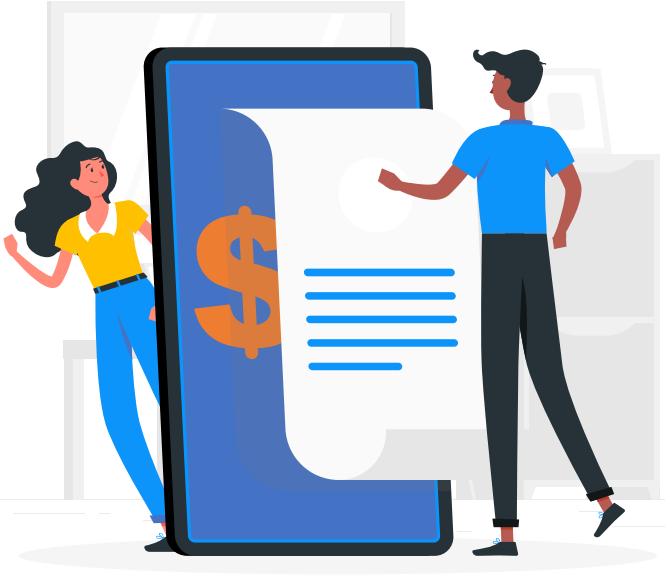
Considering what has been said so far, it's clear that the DBA figure has a different role in the cloud

- Troubleshoot Outages
- Patching
- Backups
- Install, config SQL
- Install, config OS

- Platform Architecture
- Cost Management
- Performance Optimization
- Configure Scaling

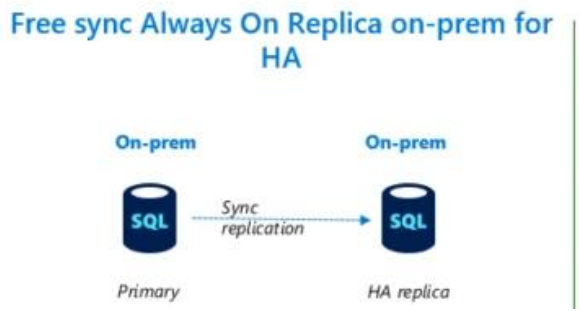
Azure Specific Tips

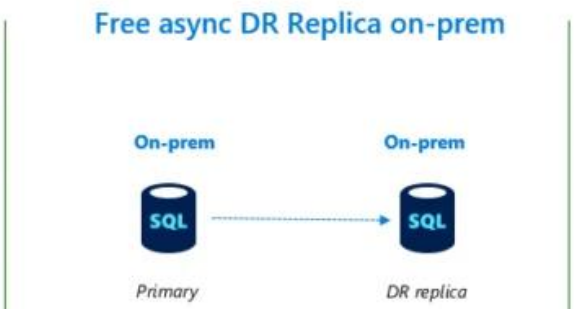
- Use DEV/Test pricing for, well, your dev/test resources
- If you have an intermittent workload, take notice of the Burstable VM types (B-Series)
- Azure does not delete virtual disks when you delete a virtual machine. Disks continue living and incurring costs until you identify them and delete them
- Microsoft is supposed to Price-match any other Cloud vendor for the same service every 3 months, if they haven't done so, try and ask why ☺

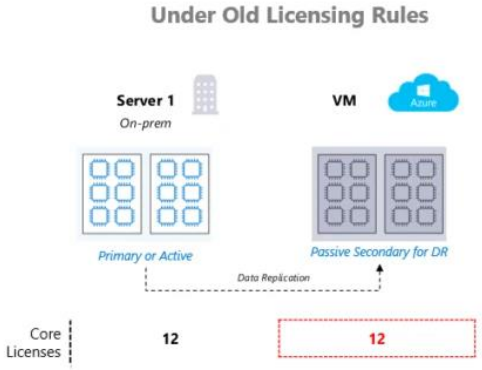


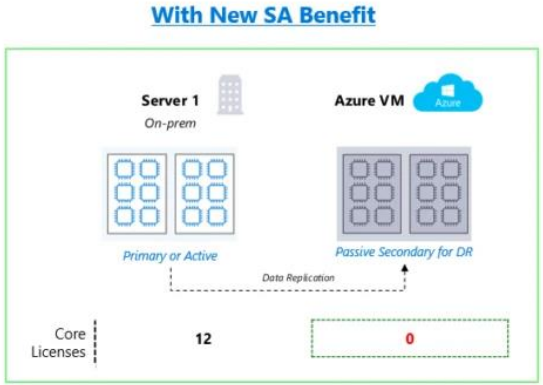
Take advantage of Licensing

Licensing changes in SQL Server introduced in the 2019 version, but applicable to all SQL versions, include the following benefits (with Software Assurance):









In addition to the high availability and disaster recovery benefits, the following operations are allowed on the passive replicas:

- Database consistency checks
- Log backups
- Full backups
- Monitoring resource usage data

The cloud

Recap

- Have a concrete goal for the move or your new project
- Measure every aspect of your current workload
- There's no magic wand, you have to measure and plan all the time
- Plan for simplicity and modularity
- Right Sizing is the key for most of it, both when deploying and scaling
- Don't be afraid of killing off resources, nothing is forever
- Reserve when it makes sense
- Keep performance tuning at the forefront
- Be aware of where your data physically is
- Data movement is not always free
- Exploit Cloud features with your apps, which must be cloud-ready

Contact me for further questions

Emanuele Meazzo

DBA, BI Facilitator, Azure Data Engineer, gets anxious when not optimizing something

Contact Info:

- Email: emanuele.meazzo@gmail.com
- Blog: tsql.tech
- Linkedin: /emanuelemeazzo
- GitHub: github.com/EmanueleMeazzo/tsql.tech-Code-snippets