

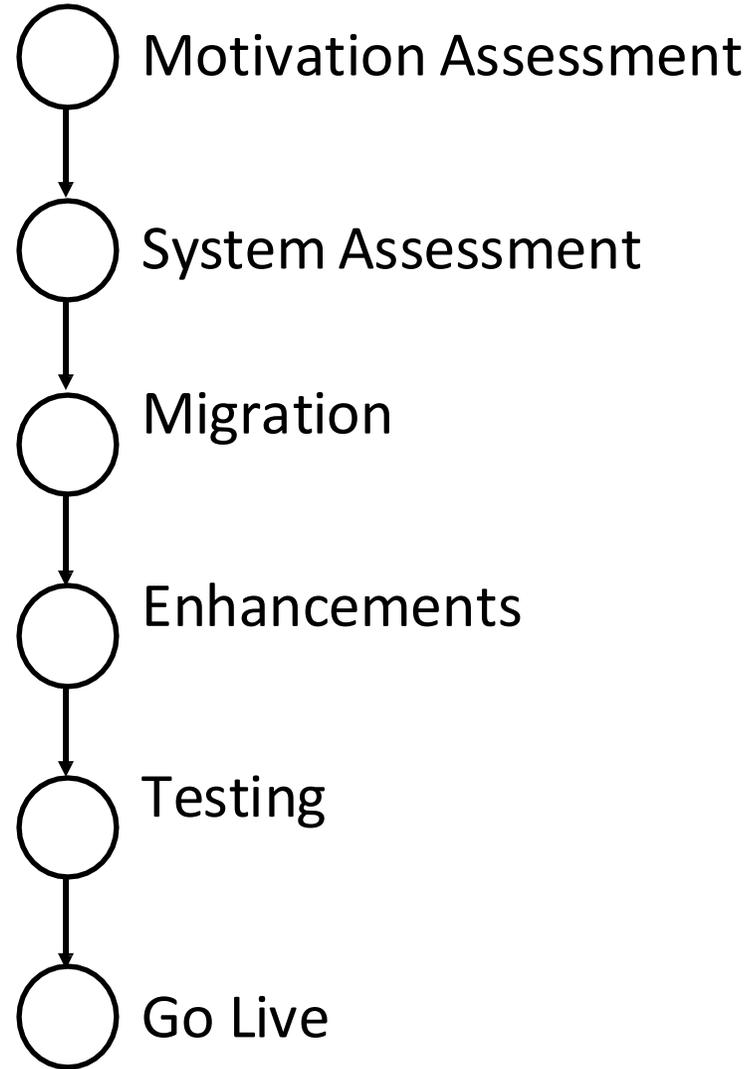


Migrating to Azure

Migrating to Azure

- Many organizations want to migrate their apps to the cloud
- Various motivations
- Has to be planned ahead, and done methodically
- Rushing to the cloud could end up badly
- There's a well defined process that should be followed in order
for the migration to complete successfully

The Migration Process



Motivation Assessment

- Answer the grand question:

Why?

(do we want to migrate to the cloud)

Possible Answers

- To save costs
- To take advantage of cloud capabilities (services, redundancy, scalability, etc)
- To modernize the system
- To attract new employees

Saving Costs

- Not always the case
- Sometimes cloud system can cost MORE than on-premises system
- CapEx vs OpEx
- A very thorough cost estimation should be conducted

Take Advantage of Cloud Capabilities

- Great motivation! (probably the best...)
- Make sure the migrated system can actually use cloud capabilities
- Example: Legacy, single-session app that cannot be scaled will end up on a single VM, just like on-prem

Modernize the System

- Usually, Modernization = Rewrite
- Why?
- Possible answers:
 - Not supported platforms
 - Difficult to hire employees
 - Licensing costs

Attract New Employees

- Not a very good motivation, but it works...
- The organization has to be part of the cloud trend in order to attract new employees

Migration Strategies

Lift & Shift



Install the app in the cloud “as-is”, without any changes, usually on a VM

Refactoring



Modify the app a bit so it can run on managed services (ie. App Services)

Rewrite



Rewrite the app from scratch, usually in a modern platform, and take advantage of all cloud services

Lift & Shift

On-Premise



VM
Java App



VM
Oracle DB

Cloud



Refactor

On-Premise



VM
nodeJS App



VM
MySQL DB

Cloud



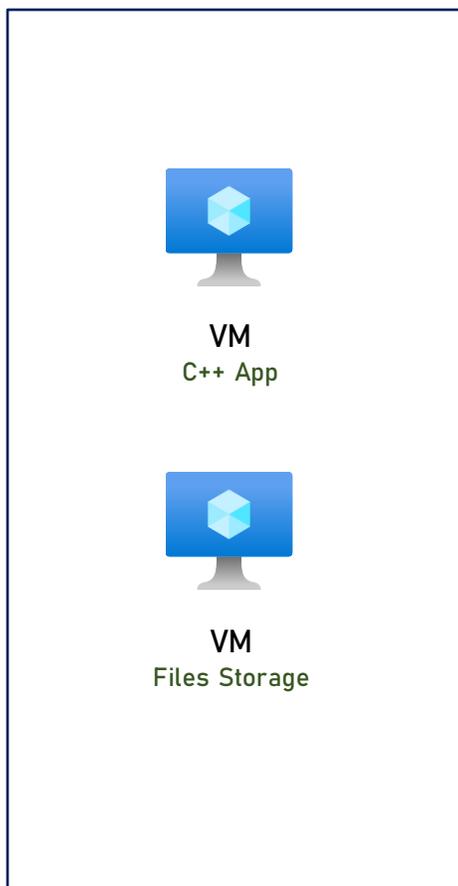
App Service
nodeJS App



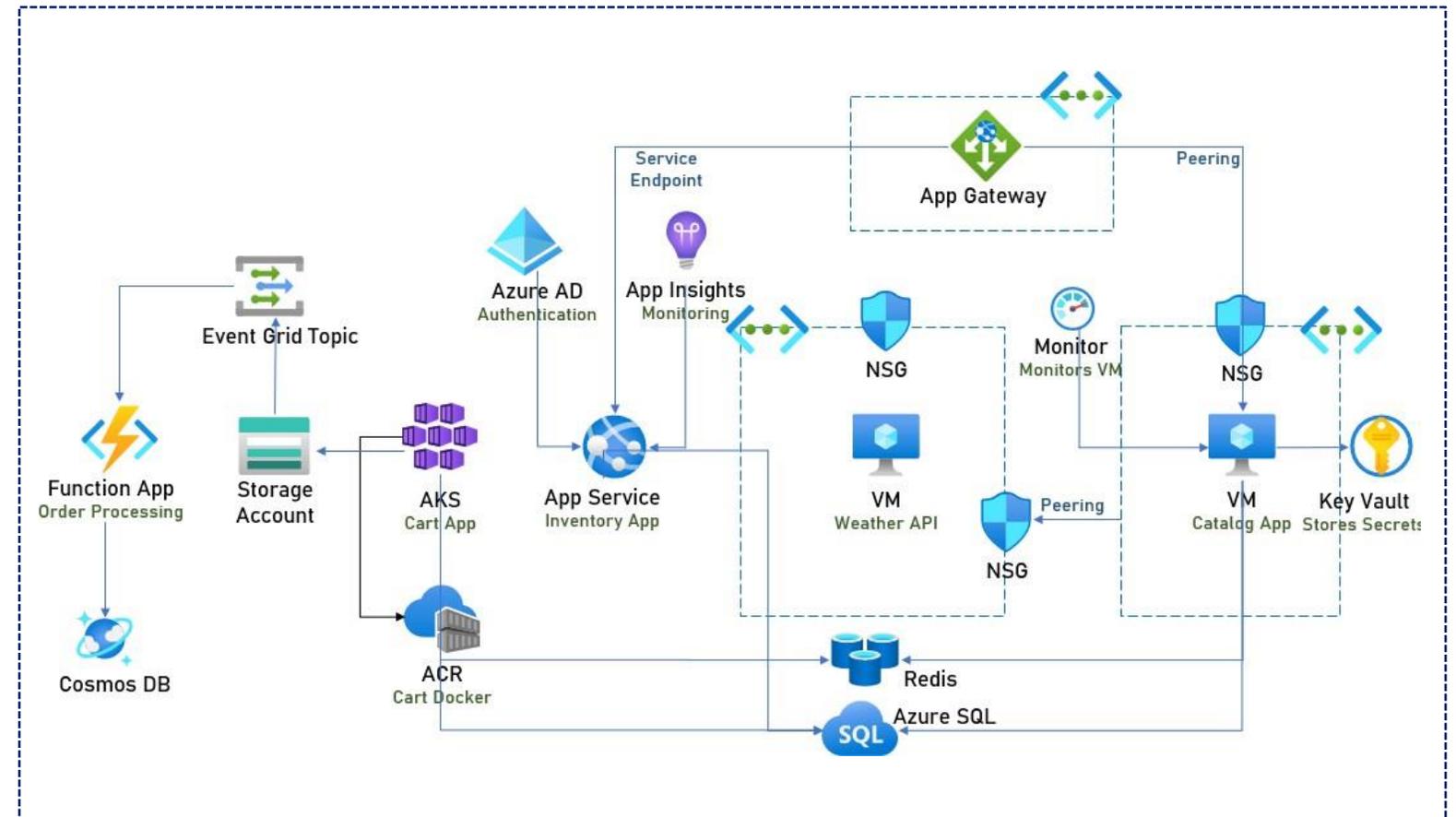
Azure MySQL

Rewrite

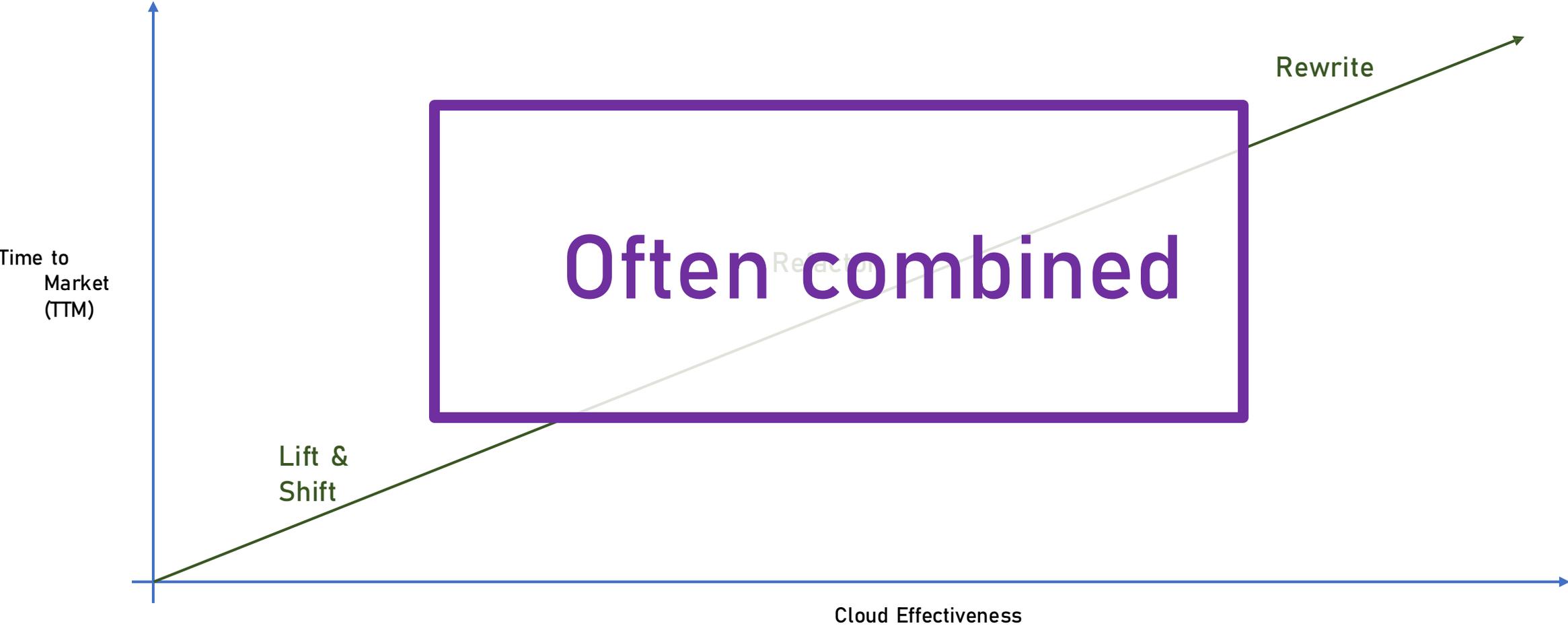
On-Premise



Cloud



Comparing Cloud Migration Strategies



System Assessment

- Goal:
 - To find out what is the best way to migrate the system to the cloud
 - Doing this by asking the right questions

System Assessment

- Questions we need to ask:

What is the runtime platform of the system and on which OS? (.NET, Java, C++ etc.)

How is it deployed today? (VM, Docker...)

What database(s) is used?

What is the authentication mechanism?

Any interfaces to
organizational

Any work with hardware? (dongles...)

System Assessment

- Answers dictate migration strategy, if possible at all:

What is the runtime platform of the system and on which OS? (.NET, Java, C++ etc.)



What is the authentication mechanism?

How is it deployed today? (VM, Docker...)

Any interfaces to organizational

What database(s) is used?

Any work with hardware? (dongles...)

System Assessment

- Answers dictate migration strategy, if possible at all:

What is the runtime platform of the system and on which OS? (.NET, Java, C++ etc.)

What is the authentication mechanism?

How is it deployed today? (VM, Docker...)

Any interfaces to organizational

What database(s) is used?

Any work with hardware? (dongles...)

System Assessment

- Answers dictate migration strategy, if possible at all:

What is the runtime platform of the system and on which OS? (.NET, Java, C++ etc.)

What is the authentication mechanism?

How is it deployed today? (VM, Docker...)

Any interfaces to organizational

What database(s) is used?

Any work with hardware? (dongles...)

Azure Migrate

- A central hub for assessing and migrating various resources
- Uses platform-specific agents to discover the resources that can be migrated
- Provides migration recommendations
- Free

Azure Migrate

The screenshot shows the Azure Migrate website interface. At the top, there is a blue navigation bar with the Microsoft Azure logo and a search bar. Below the navigation bar, the page title "Azure Migrate" is displayed with the Microsoft logo. A search bar is present, and there are links for "Get started" and "Explore more". The left sidebar contains a navigation menu with sections: "Overview", "Migration goals" (including Servers, Databases, VDI, Web Apps, Data Box), "Manage" (including Discovered items), and "Support + troubleshooting" (including New support request). The main content area features a heading "Migrate your on-premises datacenter to Azure" with a sub-heading "Discover, assess and migrate your on-premises applications using Microsoft or third-party tools, or find an expert to help with your migration. Learn more". Below this, there are three columns of content, each with an icon, a title, a description, and a button:

- Windows and Linux servers**: Discover, assess and migrate your on-premises VMware and Hyper-V virtual machines or Physical servers to Azure. Button: [Assess and migrate servers](#)
- SQL and other databases**: Assess and migrate your on-premises databases to Azure SQL Database Managed Instance or Azure SQL Database. Button: [Assess and migrate databases](#)
- Explore more scenarios**: Assess and migrate web apps, migrate data and assess virtual desktop infrastructure (VDI). Find guidance on various migration activities. Button: [Explore more](#)

Azure Migrate

- Good for low-level assessment (platforms used etc.) but won't deal

with other considerations

- ie. Interfaces with organizational systems
- Can be used as a first-step, but must perform manual assessment afterwards

Migration

- Time for actual migration...
- Migration strategy depends on the system assessment results

Lift & Shift

- The easiest strategy
- Simply create VM images and build new VMs in the cloud
- Don't forget to put VMs in appropriate VNet and define NSG

Refactor

- Should be done in two phases:
 - Lift & Shift – Make sure the app works in the cloud as-is
 - Refactor – gradually improve the system and integrate cloud services
- Don't try shortcuts!

Rewrite

- Not really a migration...
- Design the system from the ground up to be cloud native
- Use as many cloud services as possible

DB Migration

- In general:
 - Prefer the managed version of the DB (and not on a VM)
 - Migration is better done using the DB's own tools and not via

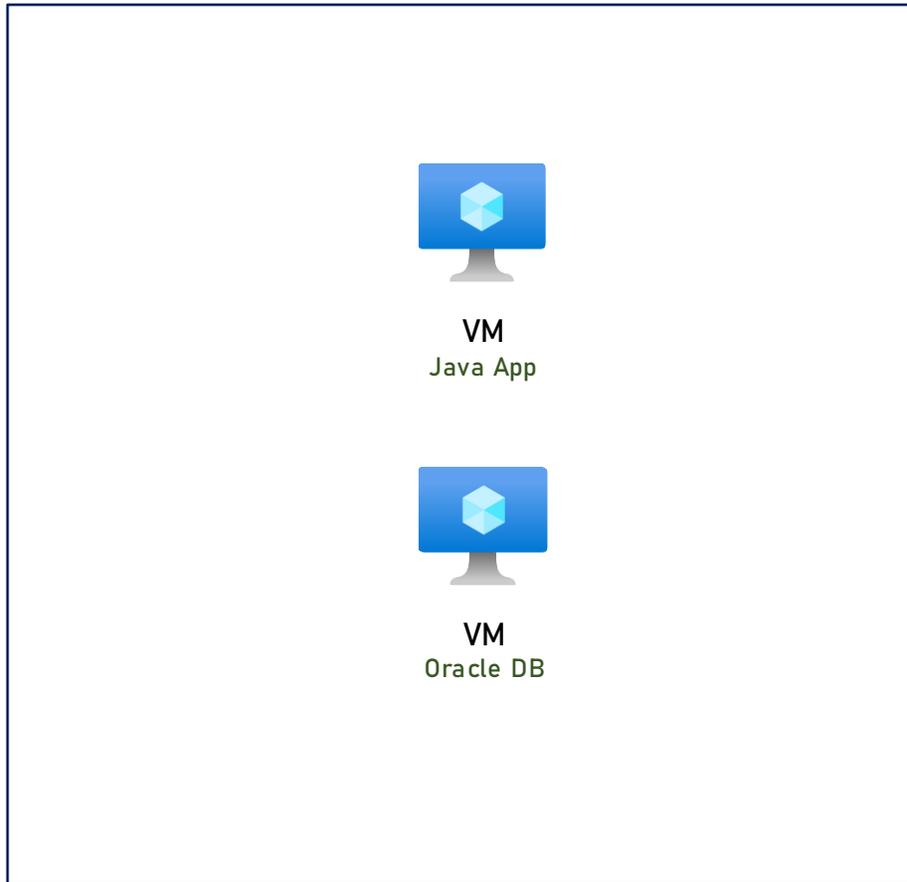
Azure migration services

App Enhancements

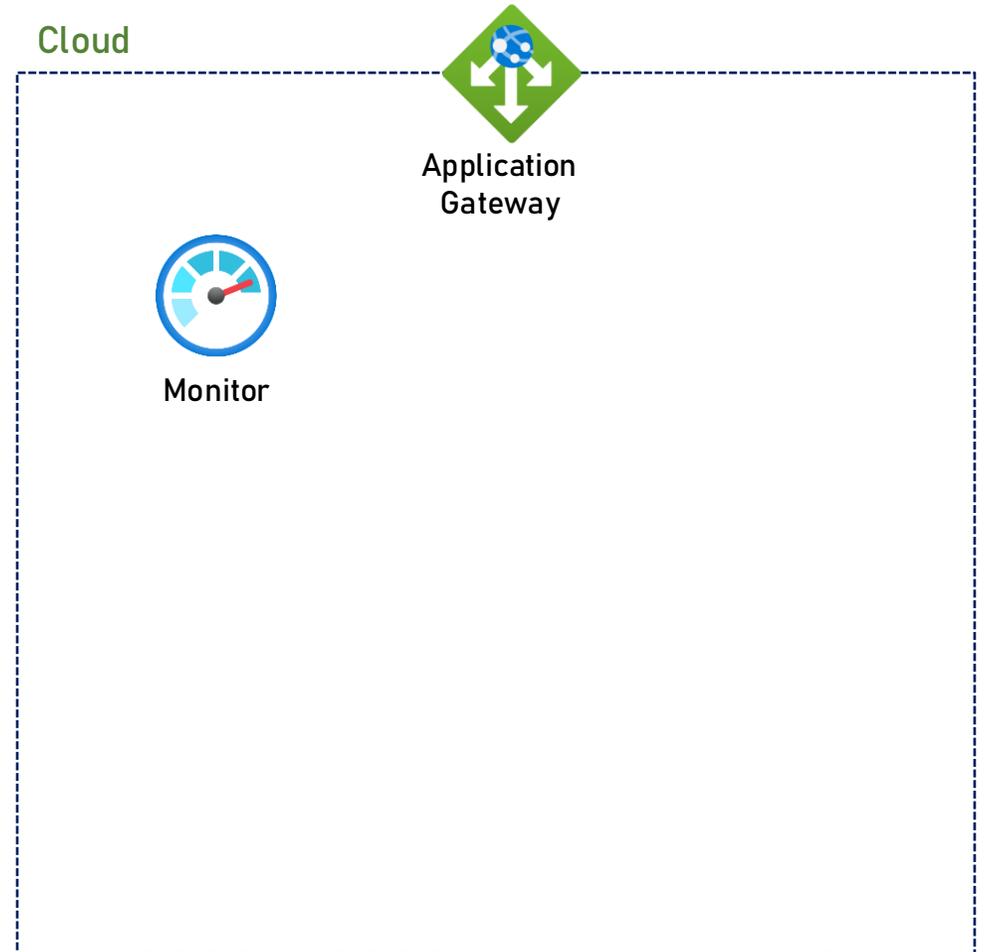
- After migration, enhance the system
- Not necessarily requires code changes
- Main areas for improvements:
 - Logging and Monitoring using Azure Monitor
 - Network protection using Application Gateway

Lift & Shift Example

On-Premise



Cloud



Testing

- Testing in cloud is similar to on-prem
- Put strong emphasis on logging and monitoring
- Make sure you have access to system's data
- If system is auto-scaled or DRed – test these scenarios
- Check performance – might vary from the on-prem figures

Go Live

- Congratulations!
- Keep an eye on the costs
 - Set up budget alerts
 - Define tags
 - Look at the data at least once a month