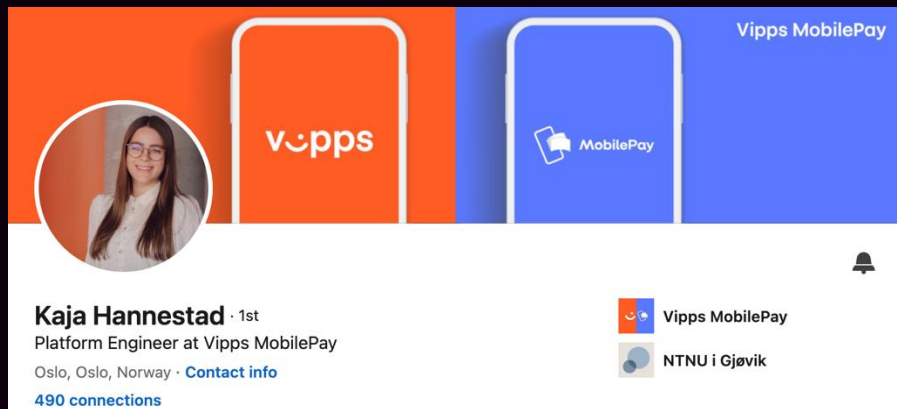


Kaja Hannestad  
Sven Malvik

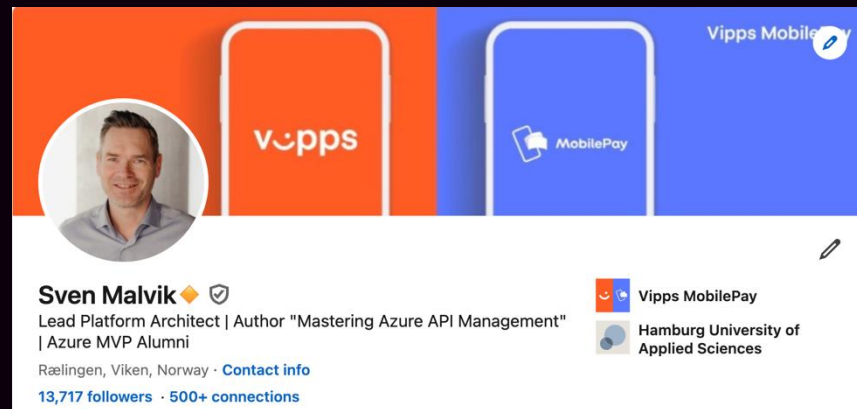
Conquering FinOps at Vipps MobilePay

# Why are Kaja and Sven giving this talk?



LinkedIn profile of Kaja Hannestad. The header features a background with the Vipps logo on an orange square and the MobilePay logo on a blue square, with the text 'Vipps MobilePay' in the top right. Kaja's profile picture is on the left. The main text reads: **Kaja Hannestad** · 1st  
Platform Engineer at Vipps MobilePay  
Oslo, Oslo, Norway · [Contact info](#)  
490 connections

Below the main text are two logos: Vipps MobilePay and NTNU i Gjøvik.



LinkedIn profile of Sven Malvik. The header features a background with the Vipps logo on an orange square and the MobilePay logo on a blue square, with the text 'Vipps MobilePay' in the top right. Sven's profile picture is on the left. The main text reads: **Sven Malvik** ♦️  
Lead Platform Architect | Author "Mastering Azure API Management"  
| Azure MVP Alumni  
Rælingen, Viken, Norway · [Contact info](#)  
13,717 followers · 500+ connections

Below the main text are two logos: Vipps MobilePay and Hamburg University of Applied Sciences.

# Agenda

1. Past
2. Present
3. Results
4. Future



## #NoWorries

I dare to fail and share my fuckups.

When the going gets tough, my generosity gets going.

I'm open by default - I share knowledge and help my colleagues succeed.



## #OneTeam

"That is someone else's problem" is not in my vocabulary.

We create dazzle together – the more complex the task, the more different skills I involve.

I'm curious about my colleagues and appreciate different points of view.



## #KeepItSimple

I speak and communicate informally and clear.  
(No corporate bs, please!)

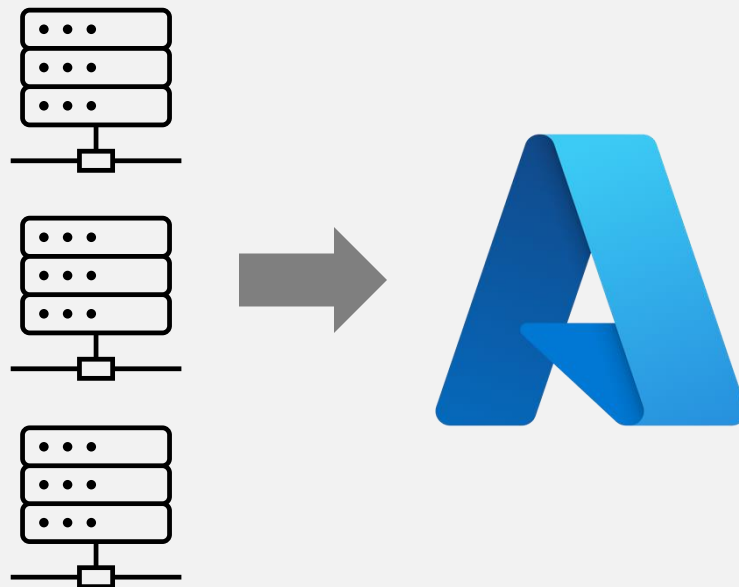
When I prioritize, customer value comes first.

I take the direct route to solve problems.



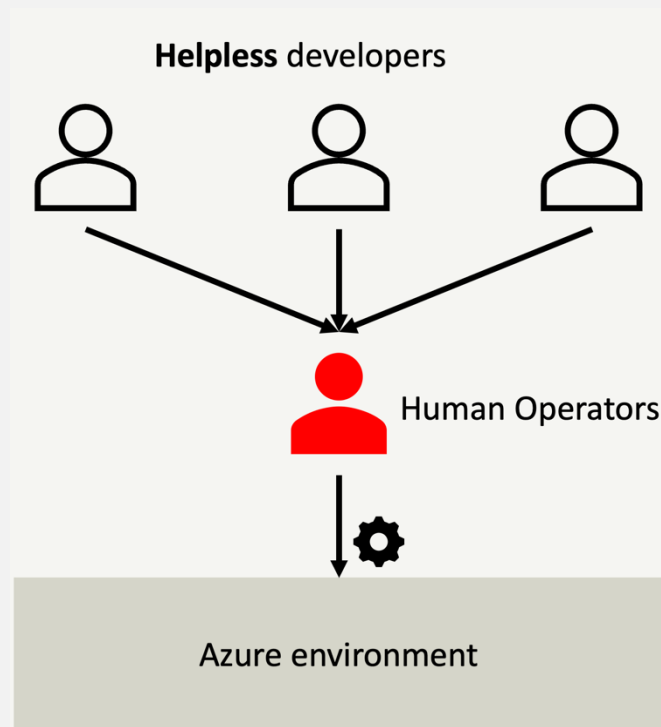
## “2018” – start of our Azure journey

- On-premise in the cloud
- Little Azure expertise
- Autonomous teams
- Focus on growth



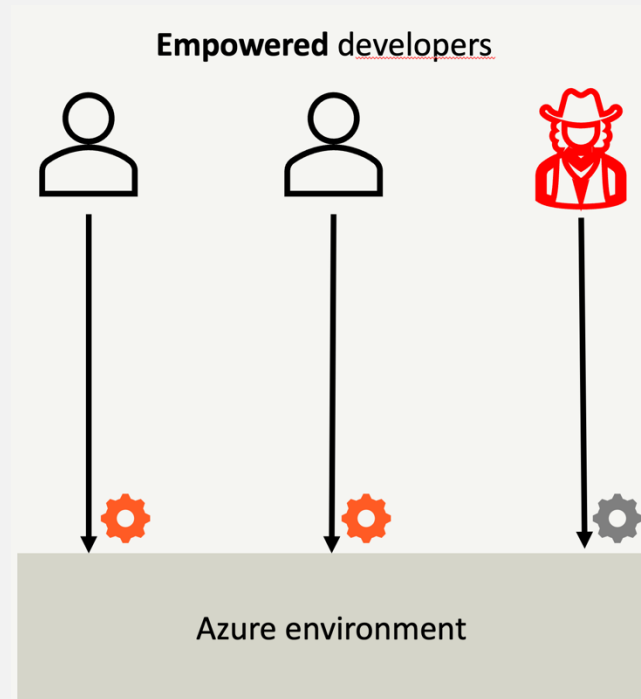
# Everything was centralized

- High degree of dependency
- Many support cases
- Not scalable



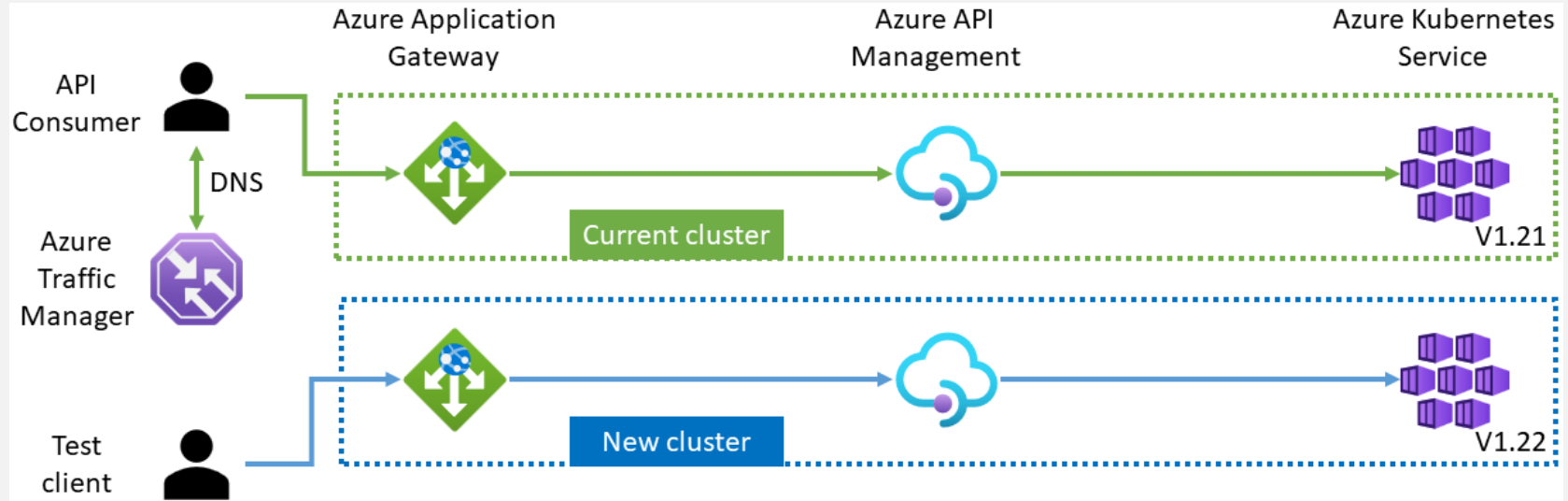
# Everyone got empowered 🤔

- High degree of autonomy
- Still many support cases
- Duplicate infrastructures





# Immutable infrastructure



The “de”-merger happened.




bank  
**axept**

The merger happened.



# The bi-product of this is ...


- From 1 to 4 countries
- From 350 to 800 people
- Different cultures meet
- Little self-service
- Focus on “One-Platform”
- More technical debt

**Sven Malvik** • You  
Lead Platform Architect | Author "Mastering Azure API Managemen...  
1mo • Edited •

WE DID IT!!!

A bit over a year ago, **Vipps** and **MobilePay** became **Vipps MobilePay**. From there we started all the preparation that was necessary for this one night. We moved MobilePay from the Danske Bank data center into **Microsoft #Azure**, or more specifically into the Vipps Core Engine; we just call it VCE – which is in very, very short Azure **#Kubernetes** Service (**#AKS**), paired with Azure App Gateway and Azure API Management (**#APIM**).


4,5 million Danish MobilePay users are now also using the same mobile app as the Norwegians who use the Vipps app – same app, different color. As I'm working directly on the Azure platform, I'm super thankful for having this amazing people in my team. A big thanks also to our contacts in Microsoft who gave us advice and also where present during the "cutover". Reach out to me if you want to know more about our journey in Azure.

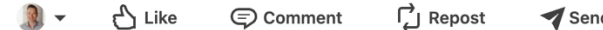
**MobilePay dropper Danske Bank og skifter til Microsoft Azure: "Det er uden tvivl den mest komplicerede..."**  
computerworld.dk • 1 min read


Marte Horne and 639 others

20 comments • 1 repost

Reactions





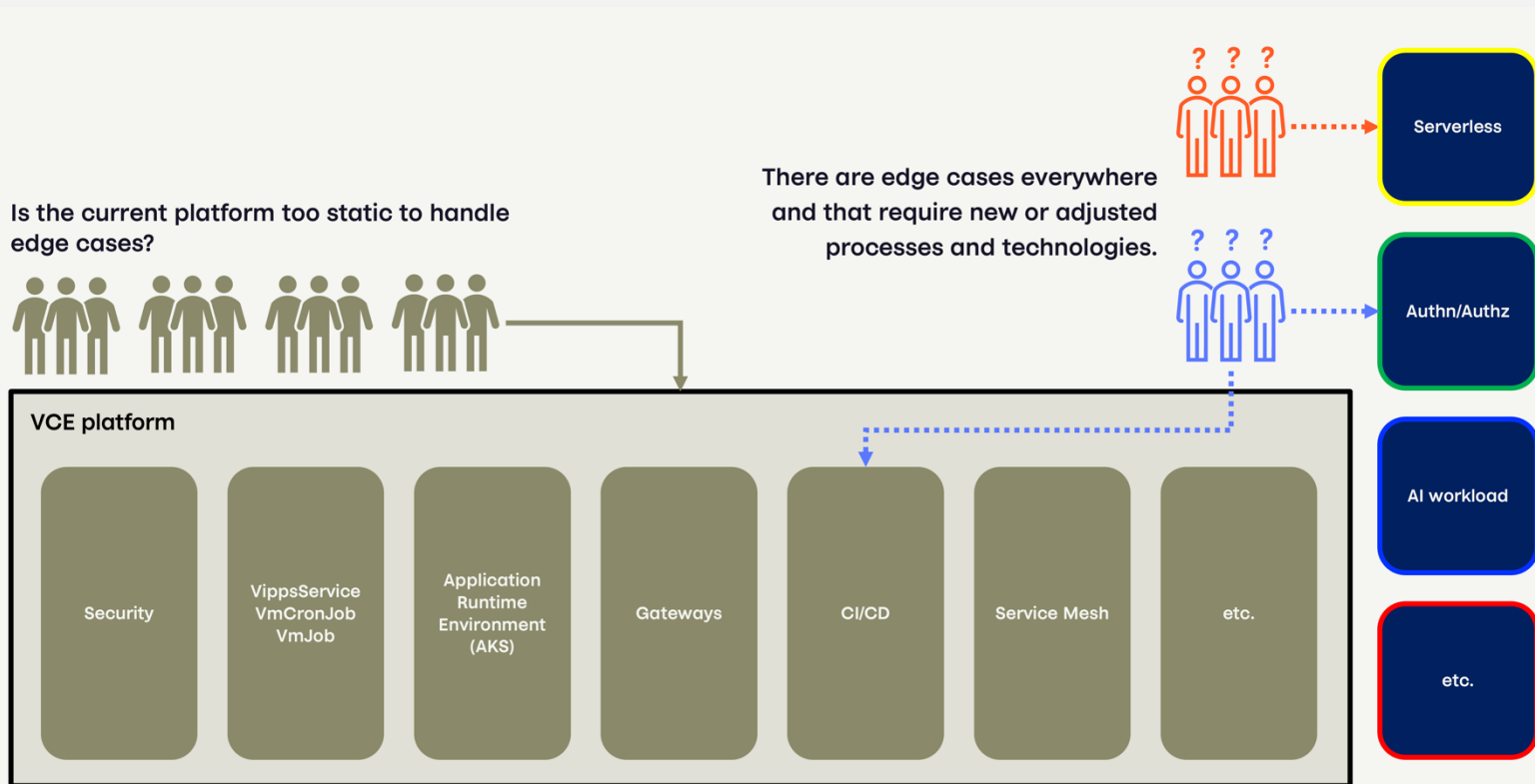
 52,585 impressions

View analytics

Our focus in Platform was stability and security, not  
our spendings or the teams



# We were focusing on something.



# Present Mapping







We closed the Lithuania office that  
we inherited from Danish MobilePay.

**E24** | Norges største næringslivsavis

Børs Aksjelive Tips oss! Kampanje Meny

## Vipps nær 100 millioner i underskudd

Produktet har blitt så populært at det har blitt et verb i det norske språk - men til tross for at vel 75 prosent av alle nordmenn «vippser», opplever Vipps røde tall.



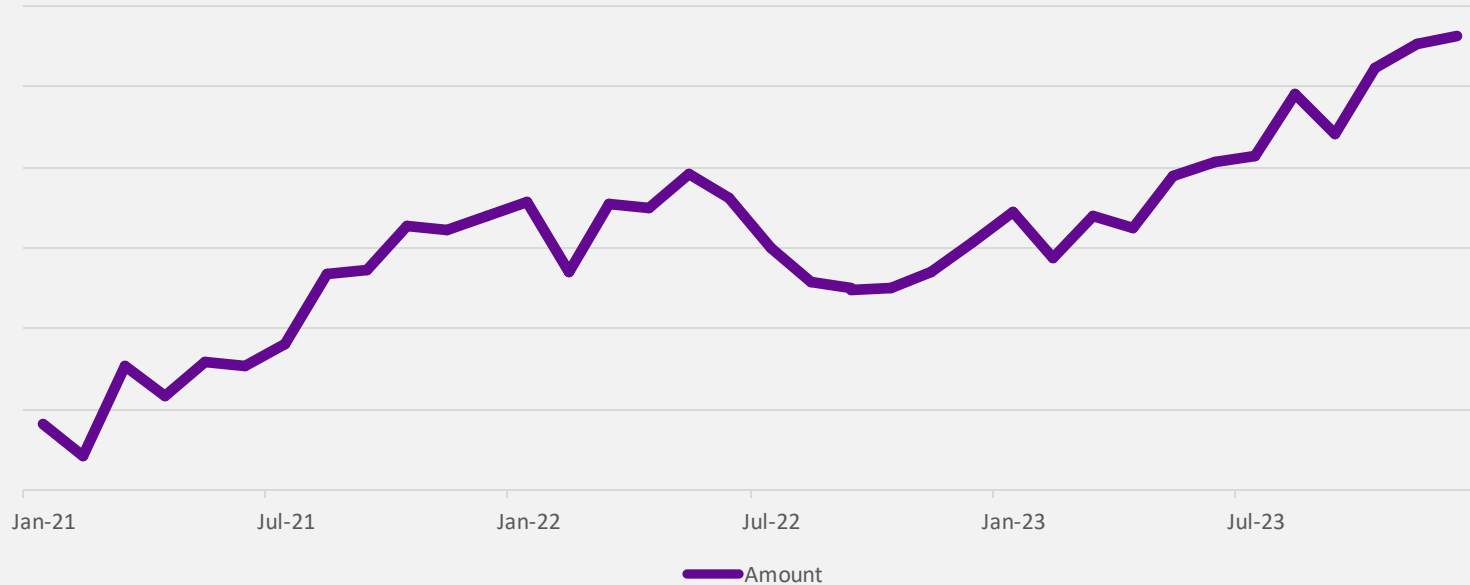
## Vipps tapte én milliard: Sparker 100 folk, kutter konsulenter

– Å legge ned et kontor og redusere bemanningen med 100 kolleger er krevende.



# Actual Azure cost past 3 years

Consumption/amount per month 2021 - 2023



- 80% increase in monthly cost from Jan '21 to Dec '23

# Azure Cost Optimization project

1

Support smooth onboarding of customers from Denmark to Azure.



2

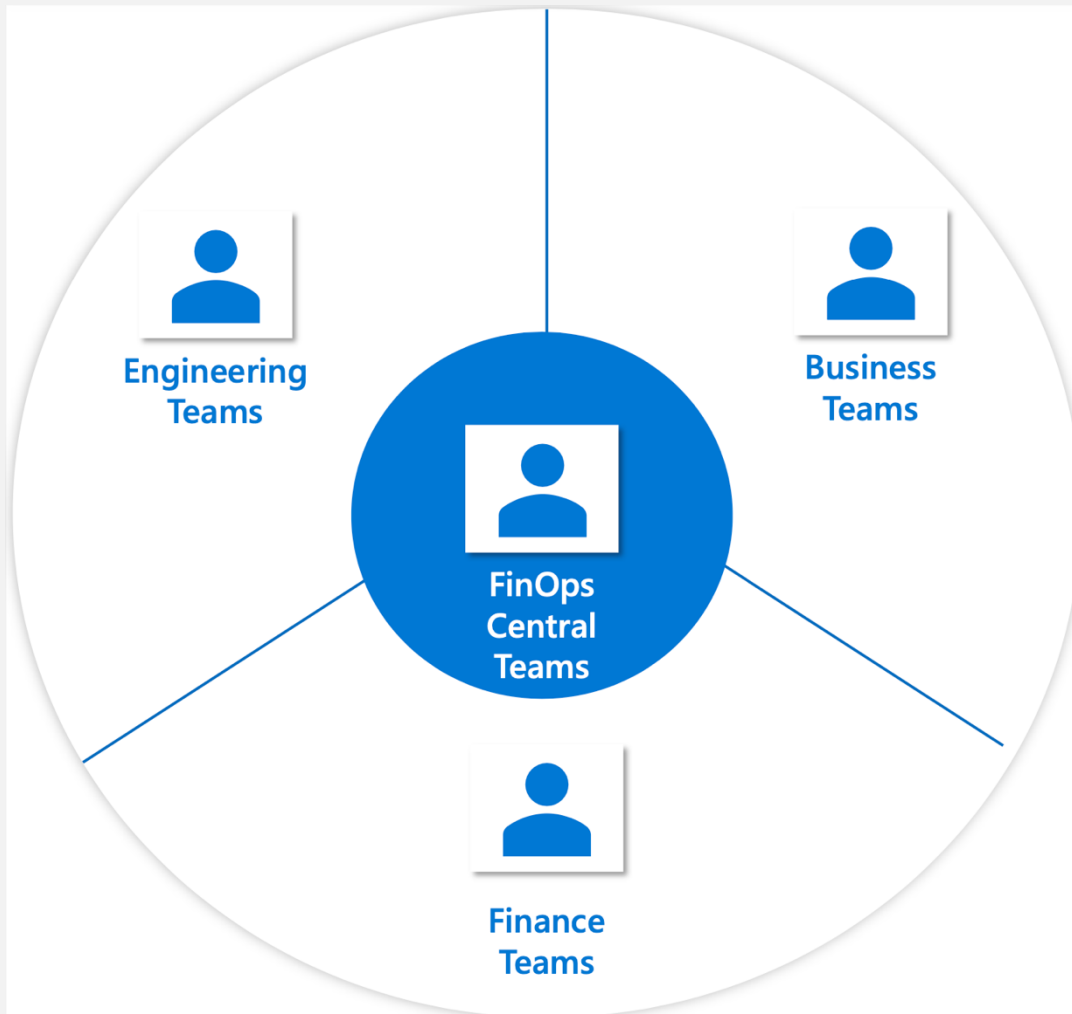
Ensure the platform is fit-for-purpose and can sustain the added load.



3

Identify cloud cost optimizations.





## FinOps Central Team

- B** usiness Teams:
  - Budget Resource Control
  - Lead Cultural Shift
  - Measuring Cloud Business Value
  - Define and manage KPIs
- E** ngineering Teams:
  - Cloud Cost Optimization
  - Enable tracking and visibility of resource utilization
  - Identify usage and spend anomalies
  - Cloud governance and management
- F** inance Teams:
  - Cloud Cost Reporting
  - Contractual model optimization
  - Negotiate pricing and commitment
  - Forecasting and predict spend

# Foundational initiatives

(increase transparency)

- **Update Azure Resource information**  
*e.g.: product catalog/team matrix, (subscription) naming conventions, (ownership) tagging applied, policies applied*
- **Allocating resources** dedicated correctly to products  
*to facilitate optimization by the Tech and Product teams*
- **Allocating shared resources** and forecasting  
*to facilitate resource and rate optimization by the platform teams*
- **Cost Optimization Policies**  
*e.g.: resource restrictions and alert/budget policies*

# Transparency matters

- All stakeholders need an easy way to see what their products cost.
- The Azure cost management plane is a good start. To better compare data, Power Bi might be a better choice.
- Shared costs.
- Tagging of resources.



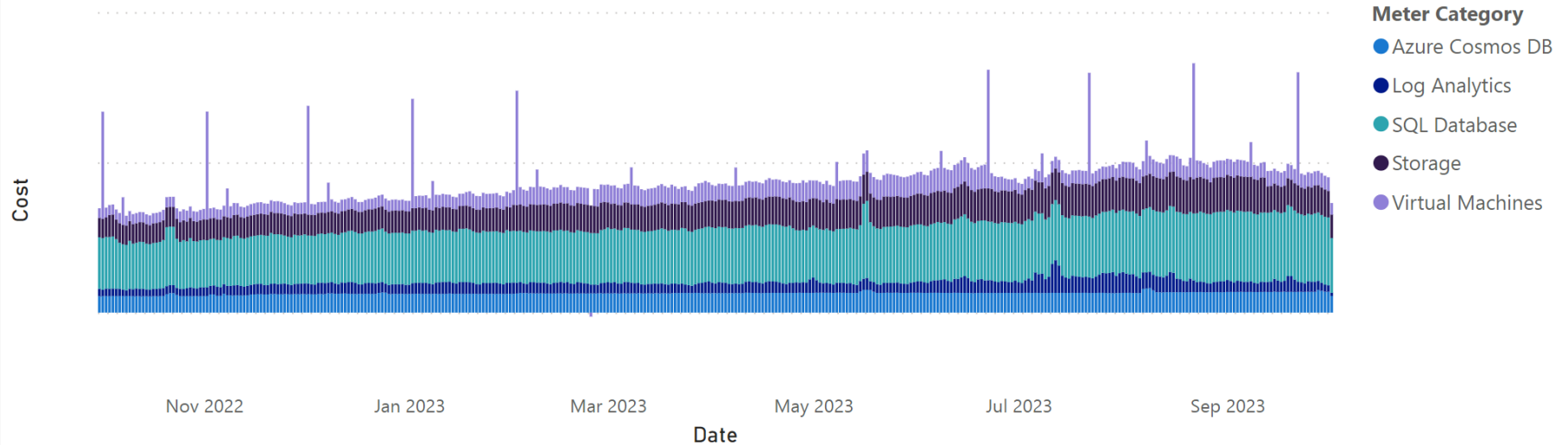
# Optimization initiatives

(direct cost optimization)

- Drive efficiency on **Dev/Test** resources  
*(e.g.: working with expiration dates and auto stop/start schedules)*
- Optimizing **inactive** resources  
*By Azure advisor (% of use), Azure monitor, retention policies*
- Optimizing **product related** resources  
*Starting with a candidate tech-product team. Preferably the most cost consuming team*
- Optimizing **shared** resources by using **reservation plans**  
*Base on the forecasting input from allocated shared resources. E.g. Independent Cosmos DB and VM's in AKS*
- Rightsizing **databases**  
*Starting with a candidate set of database. Preferably the most cost consuming databases*
- Rightsizing **Storage**  
*Starting with a candidate set of Storage (accounts). Applying Lifecycle Policies.*
- Rightsizing **VMs**  
*Starting with a candidate set of VMs.*
- Rightsizing and categorizing **Log's**  
*E.g., Log analytics and data retention policies*

# The big 5 in Power BI

Cost by Date and Meter Category





# Present Actions

# Meeting the teams

- Priority #1
- Gamified
- Budgets and alerts
- Time consuming, but effective

Team name	Accountable contact	Subscription(s)	AKS namespace(s)
-----------	---------------------	-----------------	------------------

## TEMPLATE FOR FINOPS ASSESSMENT PER TEAM

Topics	PowerBI report	Steps	Key takeaways	Actionpoints
	<u>Azure Advisor Recommendations</u>	<ul style="list-style-type: none"> <li>Any cost advisor recommendations for the team's subscriptions? <ul style="list-style-type: none"> <li>Adjust filter to 1 year term and 60 days look-back period.</li> </ul> </li> </ul>		
	<u>Orphan Workbook</u>	<ul style="list-style-type: none"> <li>Any orphan or unused resources for the team's subscriptions?</li> </ul>		
	<u>SQL</u>	<ul style="list-style-type: none"> <li>Does the team have any DBs? (SQL or Cosmos)</li> <li>Are they in use at all? <ul style="list-style-type: none"> <li>If yes, can they tolerate downtime while migrating to a shared pool? <ul style="list-style-type: none"> <li>If no to pooling, should they have a reservation?</li> </ul> </li> </ul> </li> <li>How is the utilization of these? <ul style="list-style-type: none"> <li>Do they seem to be right-sized?</li> </ul> </li> <li>How long is the retention on backup?</li> <li>Are they using SQL Audit logs? <ul style="list-style-type: none"> <li>If they are shipping to CSRT, can we turn it off?</li> </ul> </li> </ul>		
	<u>Virtual Machines</u>	<ul style="list-style-type: none"> <li>Does the team have any VMs? <ul style="list-style-type: none"> <li>What are they used for, and could it be running in AKS?</li> </ul> </li> <li>Are they in use at all?</li> <li>How is the utilization of these? <ul style="list-style-type: none"> <li>Do they seem to be right-sized?</li> </ul> </li> <li>Should they have a reservation?</li> <li>Is auto-shutdown enabled, or could it be?</li> </ul>		
	<u>App Services</u>	<ul style="list-style-type: none"> <li>Is the team running any App Services? <ul style="list-style-type: none"> <li>Are they in use at all?</li> </ul> </li> <li>Why are they not running in AKS?</li> <li>How much could be saved by moving to AKS?</li> </ul>		
	<u>Storage Accounts</u>	<ul style="list-style-type: none"> <li>Does the team have any Storage Accounts? <ul style="list-style-type: none"> <li>Are they in use at all?</li> </ul> </li> <li>Do they have lifecycle policies set with retention? <ul style="list-style-type: none"> <li>Does the retention make sense wrt. compliance?</li> </ul> </li> <li>Are there any append blobs? If yes, can we transform them to block blob and then archive them?</li> </ul>		
	<u>Log Analytics</u>	<ul style="list-style-type: none"> <li>Any Log Analytics workspaces in the team's subscription?</li> <li>Are they in use, or are they Default and can be deleted?</li> <li>If heavy in use, could commitment be an option to save money? <ul style="list-style-type: none"> <li>Or are there other options of solutions to use?</li> </ul> </li> </ul>		
	<u>Application Insights</u>	<ul style="list-style-type: none"> <li>Does the team have Application Insights?</li> <li>Are they using it?</li> <li>Is it possible to use another solution?</li> </ul>		
	<u>AKS</u>	<ul style="list-style-type: none"> <li>Do they have their own namespace in AKS? <ul style="list-style-type: none"> <li>If so, how much does it cost? (Test and Prod) <ul style="list-style-type: none"> <li>Click customize and enable amortize</li> </ul> </li> </ul> </li> <li>Are pods right sized? (Grafana)</li> <li>Does number of replicas make sense?</li> <li>Is HPA enabled? (If not, why?)</li> </ul>		
	<u>Test vs. Prod environment</u>	<ul style="list-style-type: none"> <li>Are resources in Test subscription sized logically, or are they like Prod? <ul style="list-style-type: none"> <li>Can they be sized down?</li> </ul> </li> <li>Any replication/HA in Test that can be disabled?</li> <li>Unused resources in Test that can be deleted?</li> <li>Does everything need to be running 24/7, or could IaC to deploy when needed reduce cost?</li> </ul>		
	<u>Tagging of resources</u>	<ul style="list-style-type: none"> <li>How is the team in terms of tagging of resources?</li> </ul>		
	<u>Budgets and anomaly alerts</u>	<ul style="list-style-type: none"> <li>Does the team have any budgets set?</li> <li>Does the team have anomaly alerts set?</li> </ul>		
	<u>Splunk</u>	<ul style="list-style-type: none"> <li>How much logging for the namespace compared to number of pods?</li> <li>Can it be reduced in any way?</li> </ul>		
	<u>Other</u>	<ul style="list-style-type: none"> <li>Any other resource types that stands out?</li> <li>Classic Front Doors? Move them to front doors managed by Platform for potentially lower cost (Classic do not have edge location in Europe but premium does) and easy to manage?</li> </ul>		

# DB Hotel



## #NoWorries

- Let developers focus on what they're best at – writing code
- Platform will take care of security and best practices



## #OneTeam

- We'll save a lot of money by pooling DBs together
- Pre-defined sizes and access control will make it easier for everyone



## #KeepItSimple

- Spend less time on database creation and management
- Daily, automatic rebuild of indexes for optimization

# Create New Database in DB Hotel

Use this template to create a new database in the DB Hotel. Read more in [the docs for DB Hotel](#).

1

Please fill in the details for the database to be created

2

Please fill in additional details

3

Review

Name of database to be created\*

Required to be in the format of sqldb-<product>-<environment>, where product is a unique identifier for your product and environment is uat, mt or prod. This follows Azure [best practices](#).

Where is your DB schema file(s) stored? (Optional)

GitHub repository where your database schema is stored. This is optional, and only used for tagging in Azure.

Environment to deploy to\*

PROD

Note that the database size options and [backup settings](#) will differ depending on the environment you choose.

Size of your database\*

Small (up to 2 vCores)

If you are unsure, select the smallest size - you can always scale up later. Contact us at [#platform-support](#) if these options does not fit your needs.

BACK

NEXT

# Create New Database in DB Hotel

Use this template to create a new database in the DB Hotel. Read more in [the docs for DB Hotel](#).



Please fill in the details for the database to be created

2

Please fill in additional details

3

Review

Owner team of database\*

The team responsible for database. Do NOT select team with '-owner' suffix.

System\*

Choose a sensible system for your component. A system is a collection of one or multiple components, resources, and APIs.

Is the data to be stored confidential?\*

Confidential

Please note that the confidentiality level will affect the [access control](#) for your database. Refer to [Standard for Classification of Information](#) for more information.

Does the database contain personal data?\*

Yes

BACK

REVIEW

# DB Hotel



Elastic Pool



Autoscaling



RBAC



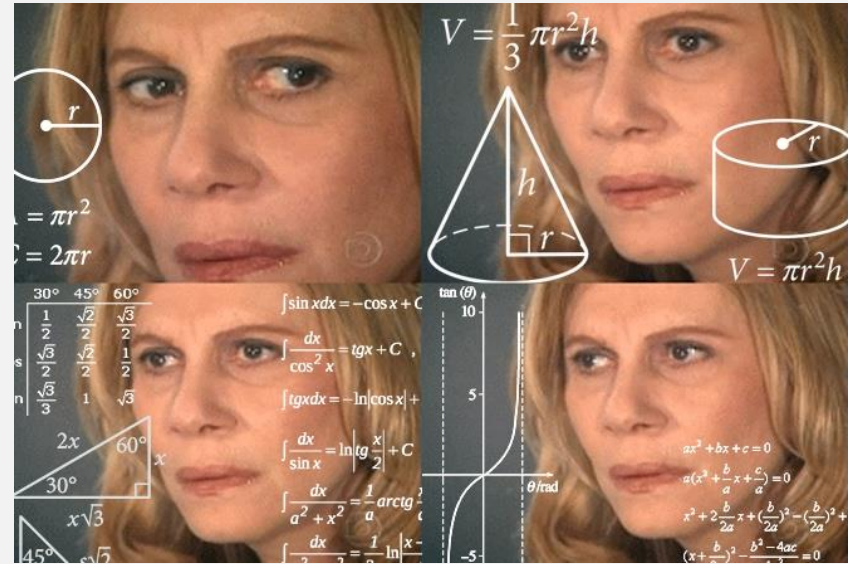
Backup



Monitoring

# Storage

- Unused
- Tiers;  
Hot, Cool, Cold, Archive
  - Cost at rest,  
read and write operations
- Lifecycle Management  
Policy



# Reservations



COMMIT TO SPECIFIC  
USAGE FOR 1 OR 3  
YEARS – GET DISCOUNT!



ADVISOR IN AZURE  
PORTAL.



RESERVATIONS ARE PER  
SKU, SO NEED TO ALIGN  
ACROSS TENANT.



LOTS TO SAVE ON  
VIRTUAL MACHINES  
(COMPUTE).



*SAVINGS PLAN.*



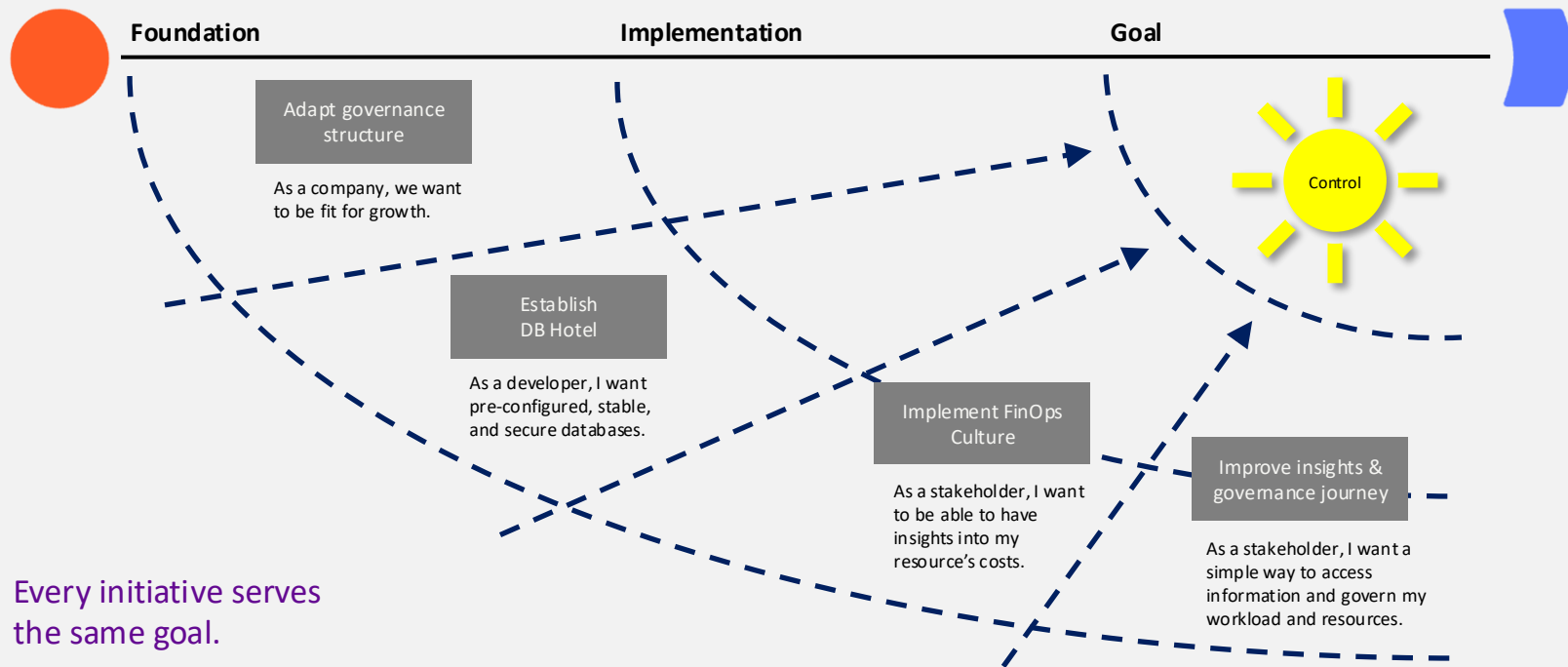
Resource	Term (months)	Lookback (days)	Estimated savings (%)
Cosmos DB - 100 RU/s	12	30/60	3%
Cosmos DB - 100 Multi-master RU/s	12	30/60	3%
Cosmos DB - 100 RU/s	36	30/60	15%
Cosmos DB - 100 Multi-master RU/s	36	30/60	15%
Azure Database for PostgreSQL Flexible Server General Purpose Ddsv4 Series Compute	36	30/60	51%
Azure Database for PostgreSQL Single Server General Purpose - Compute Gen5	12	30/60	18%
Azure Database for PostgreSQL Flexible Server General Purpose Ddsv4 Series Compute	12	30/60	27%
Azure App Service Isolated v2 Plan	36	30/60	35%
Azure App Service Isolated v2 Plan - Linux	36	30/60	45%
Azure App Service Premium v3 Plan - Linux	36	30	45%
Azure App Service Isolated v2 Plan	12	30/60	14%
Azure App Service Isolated v2 Plan - Linux	12	30/60	21%
Azure App Service Premium v3 Plan - Linux	12	30	21%
Azure Redis Cache Premium - P1 - West Europe	36	30/60	45%
Azure Redis Cache Premium - P1 - North Europe	36	30/60	45%
Azure Redis Cache Premium - P1 - West Europe	12	30/60	22%
Azure Redis Cache Premium - P1 - North Europe	12	30/60	22%
Red Hat Enterprise Linux	12	30/60	83%
SQL Database SingleDB/Elastic Pool Hyperscale - Compute Gen5	12	30/60	21%
SQL Database Single/Elastic Pool General Purpose - Compute Gen5	12	30/60	21%
SQL Database Single/Elastic Pool Business Critical - Compute Gen5	12	30/60	21%
SQL Database Single/Elastic Pool Hyperscale - Premium Series Compute	12	30	21%
SQL Database SingleDB/Elastic Pool Hyperscale - Compute Gen5	36	30/60	45%
SQL Database Single/Elastic Pool General Purpose - Compute Gen5	36	30/60	45%
SQL Database Single/Elastic Pool Business Critical - Compute Gen5	36	30/60	45%
SQL Database Single/Elastic Pool Hyperscale - Premium Series Compute	36	30/60	45%
SQL Database Single/Elastic Pool Hyperscale - Premium Series Memory Optimized Compute	36	30	45%
Azure Blob Storage - Archive LRS - 100 TB - WE	36	30/60	1%
Azure Blob Storage - Hot LRS - 100 TB - WE	36	30/60	12%
Azure Blob Storage - Cool LRS - 100 TB - WE	36	30/60	20%
Azure Blob Storage - Hot GZRS - 100 TB - WE	36	30/60	12%
Virtual Machine - Standard_F2s_v2 - WE	36	30	56%
Virtual Machine - Standard_F2s_v2 - WE	36	60	56%
Virtual Machine - Standard_DS1_v2 - WE	36	30/60	50%
Virtual Machine - Standard_E2ads_v5 - WE	36	30/60	54%
Virtual Machine - Standard_D2d_v5 - WE	36	30/60	54%
Virtual Machine - Standard_DS1_v2 - Norway East	36	30/60	50%
Virtual Machine - Standard_D2s_v3 - Norway East	36	30/60	47%
Virtual Machine - Standard_D2ds_v5 - WE	36	30/60	54%
Virtual Machine - Standard_E2ds_v4 - WE	36	60	54%
Virtual Machine - Standard_D2as_v4 - WE	36	60	44%

# Azure Policy

- Governance tool
- Automated tagging
- VM SKU's
- Restrict deployment of resources
  - Or specify allowed settings
- *Audit policies*
- *Remediation tasks*



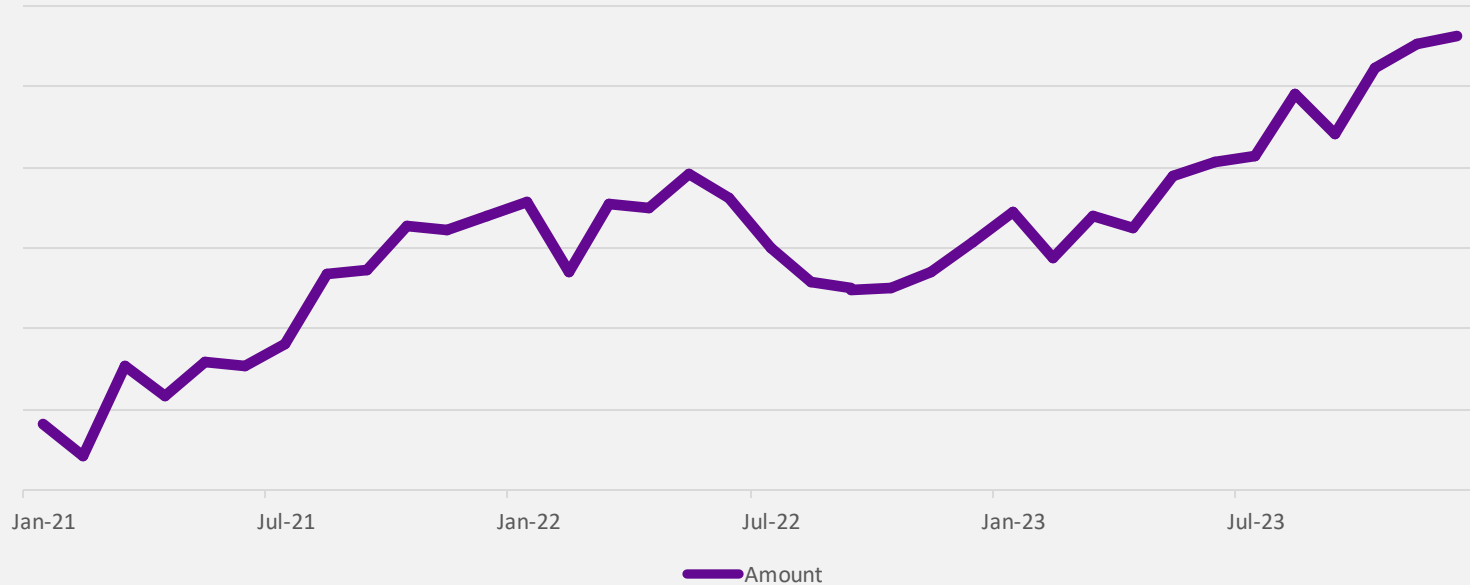
# Focusing on our FinOps culture





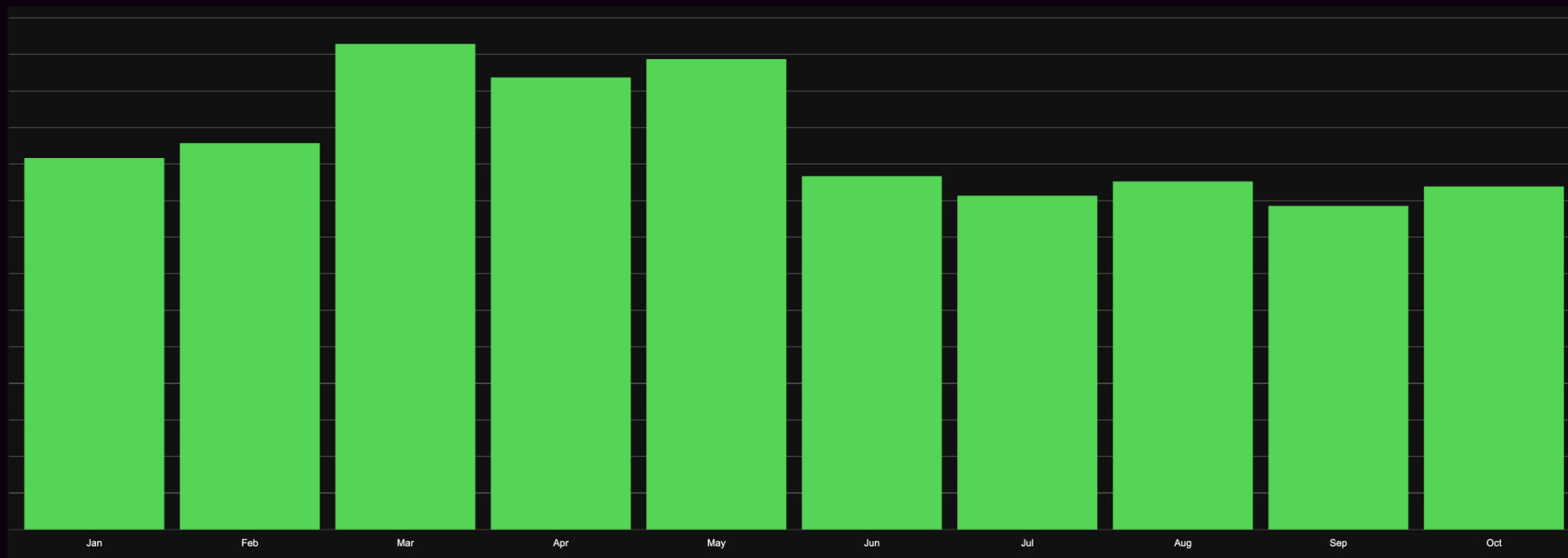
# Actual Azure cost past 3 years

Consumption/amount per month 2021 - 2023



- 80% increase in monthly cost from Jan '21 to Dec '23

# Actual Azure cost 2024



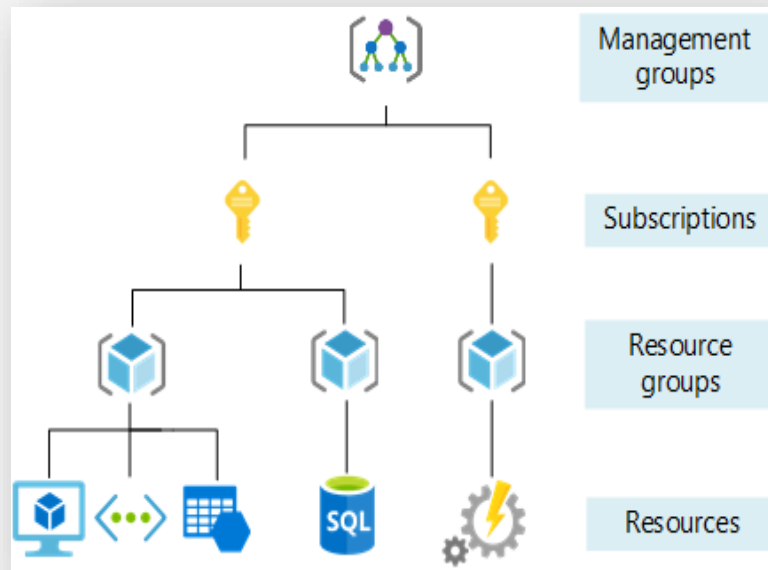
- 30% decrease in monthly cost from Mar '24 to Oct '24

# Future

## Lessons Learned

# Structure matters


- Who is accountable?
- What are the resources used for?
- What does product A, which is partially the sum of all the resources it consumes, cost?
- Why we chose to go from team- to product ownership?
- How do you tag your resources with what values?






# Lessons learned


- Culture matters
- Know your teams
- Know your resources
  - Budgets
  - Tagging
  - Self-service like DB hotel
- Teams change more often than products
  - Product-scope, not team-scope




1. nov. 2024 | Oslo  
**Join Vipps!**  
Platform Engineer  
Vipps MobilePay  
1 stilling






1. nov. 2024 | Oslo  
**Join Vipps as our new Senior Platform Engineer!**  
Senior Platform Engineer  
Vipps MobilePay  
2 stillinger





to dager siden | Oslo  
**Join Vipps as our new CRM Architect!**  
CRM Architect  
Vipps MobilePay  
1 stilling






**Siste frist**  
Oslo  
**Are you ready to shape the future online payments?**  
Product Lead Online Payments  
Vipps MobilePay  
1 stilling



Vipps MobilePay

**Kaja Hannestad** · 1st  
Platform Engineer at Vipps MobilePay  
Oslo, Oslo, Norway · [Contact info](#)  
490 connections

Vipps MobilePay  
NTNU i Gjøvik

Vipps MobilePay

**Sven Malvik** ♦  
Lead Platform Architect | Author "Mastering Azure API Management"  
| Azure MVP Alumni  
Rælingen, Viken, Norway · [Contact info](#)  
13,717 followers · 500+ connections

Vipps MobilePay  
Hamburg University of Applied Sciences