



#Milano

# Una vita da FinOps, ovvero come controllare i costi in Azure

Alessandro Scardova

Man in Finance @ Mahiz



#Milano

**improve**



**TD SYNnex**

**Grazie ai nostri sponsor**





#Milano

# Gestire i costi creando innovazione

Non “tagliare”, ma governare

Obiettivo di oggi  
→ Consapevolezza



#Milano

# Chi sono?

Sono quello che analizza i costi ed emette la **fattura** 😊

Tre domande ricorrenti che ricevo:

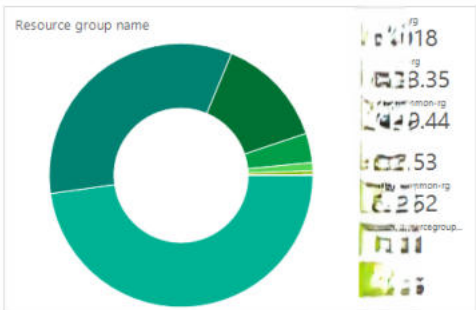
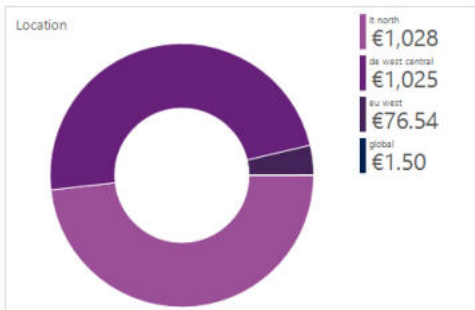
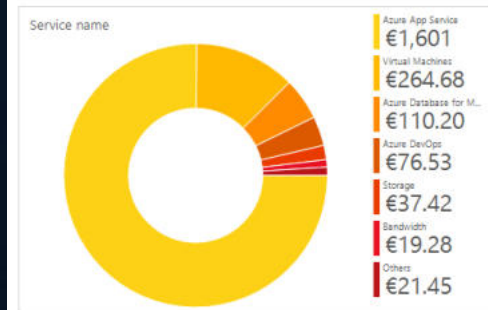
- ✓ Dove stiamo spendendo?
- ✓ Perché stiamo spendendo così?
- ✓ Puoi cambiare le cose (senza rompere la produzione)?





# Perché FinOps?

#Milano





#Milano

# Cloud = costo variabile

Nel cloud puoi creare risorse in minuti  
→ la spesa cambia in tempo reale.

Senza governance

- ✓ sprechi, duplicazioni
- ✓ sorprese a fine mese.



#Milano

# Cosa è FinoOps

Pratica culturale e **operativa**

- ✓ Governance + Finance + Engineering
- ✓ Cultura di responsabilità condivisa

Non è solo risparmio

- ✓ collega la spesa al **valore**



#Milano

# Il *loop* FinOps

Inform

→ Visibilità dei costi, licenze, utilizzo.

Optimize

→ Tariffe, utilizzo, efficienza

Operate

→ Esecuzione e miglioramento continuo





#Milano

# Azure Cost Management

Cost analysis

✓ Resource group

✓ Servizio

✓ Tag

Interrogazioni con Copilot

Esportazione dei dati per analisi (e fatture)



#Milano

# No Tag, no Party!

Tag (in minuscolo!)	Scopo	Valori di esempio
owner	Responsabile del costo	team-platform
project / app	Progetto / Prodotto	global-azure-26
environment	Ambito operativo	dev / test / prod
cost-center	Centro di costo	cc-1234
criticality	Priorità di business	high / low

Evitare tag troppo dettagliati e diversi tra i resource group



#Milano

DEMO

View march cost for tag Environment:prod

 Copilot AI-generated content may be incorrect

Here is the cost analysis for the tag 'Environment' with the value 'prod' for March 2026:

Service Name	Charges (€)	% of Total Charges
Azure App Service	1,318.00	76.13%
Virtual Machines	264.68	15.29%
Azure Database for MySQL	110.20	6.37%
Storage	19.05	1.10%
Bandwidth	15.64	0.90%
Virtual Network	3.15	0.18%
Azure DNS	0.60	0.03%

The total charges for this period were €1,731. Please note that credits and taxes are not included in these figures.

**Agentic Cloud Ops  
ore 17.15  
Giulio Sciarappa**



# Budget Alert

#Milano

Budget è una previsione di spesa per

- ✓ Subscription / Resource group
- ✓ Ovviamente Tag
  
- ✓ Fondamentale sui costi variabili



# Anomaly Alert

#Milano

Rilevano comportamenti di spesa anomali  
in base ad analisi **automatica** dello storico

Aiutano a intercettare

- ✓ Errori
- ✓ Attacchi
- ✓ Costi inaspettati

**Attenzione  
alla  
Scadenza**



#Milano

# Quota Alert

Non legato direttamente ai costi

Aiuta a capire quando si stanno esaurendo risorse



#Milano

# Azure avvisa, non spegne!

Sottoscrizioni con tetto di spesa:

- ✓ Azure for Student
- ✓ Azure Free Trial
- ✓ Azure MSDN (Enterprise e Professional)
- ✓ Vecchi piani Action Pack 😊



#Milano

# Esportare e integrare

Esportare cost & usage per

- ✓ Costruire dashboard/controlli custom
- ✓ Creare report personalizzati
- ✓ Analizzare con AI



#Milano

## 🏷️ ANALISI PER TAG

### COSTI PER OWNER

acme-spa  
dealer

**Conteggio**

36  
126

**Costo Totale**

1.050 €  
656 €

### COSTI PER ENVIRONMENT

prod

**Conteggio**

157

**Costo Totale**

1.442 €

### COSTI PER STACK

be (Backend)  
fe (Frontend)

**Conteggio**

32  
152

**Costo Totale**

842 €  
1.437 €

### COSTI PER PROMOTION REGION

it (Italia)

**Conteggio**

158

**Costo Totale**

1.686 €

### COSTI PER APP PROFILE

wordpress

**Conteggio**

24

**Costo Totale**

573 €

# DEMO



#Milano

# Provenienza dei costi

- ✓ Compute (VM/Container/PaaS)
- ✓ Storage (dischi, snapshot, backup)
- ✓ Networking (egress, firewall, bilanciatori)
- ✓ Servizi gestiti: costo variabile, richiede osservabilità.



#Milano

# Errori più comuni

- ✓ Ambienti dev / test inutilmente accesi
- ✓ Risorse “orfane” (dischi, snapshot, IP)
- ✓ Autoscaling eccessivo o mal governato.
- ✓ Tagging assente → costi difficilmente attribuibili.
- ✓ Mancanza di Alert / Budget



#Milano

# Rimedi della nonna

- ✓ Spegnere/rightsizing di risorse inutilizzate
- ✓ Ripulire storage orfano di VM cancellate.
- ✓ Tagging minimo vitale (owner, env, cost center, app).
- ✓ Autoscaling: definire limiti e metriche sensate.



#Milano

# Ridurre la spesa: Reservation e Savings Plans

Se prevedete un impegno delle risorse per 1 o 3 anni:

- ✓ Reserved Instance
  - ottima per scenari stabili
  - maggiore scontistica
  - non tutte le SKU
- ✓ Saving Plan
  - più flessibile
  - minore scontistica

**Non è la  
soluzione  
per business  
in crisi!**



#Milano

# Gli sconti non eliminano lo spreco

Passi da seguire prima di attivare una scontistica:

- ✓ Ridurre gli sprechi
- ✓ Riassegna eventuali reservation attive
- ✓ Reservation per baseline stabili e prevedibili
- ✓ Savings Plan per carichi variabili e dinamici



#Milano

# Progetti low budget / sperimentali

Se il progetto può stare fermo, deve costare zero.

- ✓ Azure Function (Consumption)
- ✓ Container App (scale-to-zero)
- ✓ Azure Container Instance (ACI)



#Milano

# Progetti a budget limitato

Qui conta più la prevedibilità che il minimo costo assoluto.

- ✓ Azure App Service (Basic / Standard)
- ✓ Azure Container App (minimo 1 replica)
- ✓ Azure SQL / servizi PaaS serverless



#Milano

# Progetti core business - critici

Costi più alti, ma prevedibili

- ✓ VM / VMSS opportunamente configurate
- ✓ Azure App Service Premium
- ✓ AKS (solo se serve davvero)

Possibilità di usare Reservation



#Milano

# Progetti altalenanti

- ✓ Azure Container Apps
- ✓ AKS
- ✓ Functions
- ✓ Da monitorare con Anomaly Alert

Elasticità senza monitoraggio = sorpresa a fine mese.



#Milano

# FinOps e AI nei team Dev

I team tendono a usare lo stesso agente / LLM per tutto

Modelli potenti non sempre necessari

Risultato:

- ✓ costi inutili
- ✓ latenza maggiore
- ✓ spreco «mascherato da produttività»



# Pricing

#Milano

Pricing Calculator

→ <https://azure.microsoft.com/en-gb/pricing/calculator>

Se vi viene voglia di fare un tool per stimare i costi:

→ <https://prices.azure.com/api/retail/prices>

Documentazione: <https://go.alle.dev/azure-prices-api-doc>



#Milano

# KPI e routine

La parte noiosa che fa risparmiare

Alcuni KPI:

→ costo per prodotto, environment, deviazione dal budget, trend  
**esempio: se dev > prod = problema da risolvere**

Routine:

→ Review periodiche con Finance e Team Dev



#Milano

# Cultura e automazione: la vera differenza

- ✓ Policy di provisioning, formazione continua.
- ✓ Automazione di tagging/compliance/cleanup (evitare regressioni).
- ✓ Inserire FinOps **fin** dall'inizio del progetto

**Non  
dopo le  
MAZZATE**



#Milano

# La checklist finale

**Inform** → visibilità dei costi

- ✓ Tag strutturati + dashboard + export dati.
- ✓ Ottimizzare
- ✓ Pulizia e resizing corretto
- ✓ Operare con disciplina



#Milano

# Q&A



Anche i feedback sono apprezzati!



alle.dev - as@mahiz.it



#Milano

# Qual è il segnale che FinOps sta funzionando?

Quando nessuno chiede più:  
"Perché la fattura è così alta?"



#Milano

**Slide e video:**

**<https://www.globalazuremilano.it>**