

Från pusselbitar till affärsnytta

Fabric, Data Agents, Copilot och Foundry

advania

Alexander Arvidsson | @alexanderarvidsson.se

Frågan är inte om vi kan prata med datan.

Frågan är inte om vi kan prata med datan.

Frågan är om svaret går att lita på.

Vilka är komponenterna?

Vad krävs för att de ska fungera ihop?

Vilka problem löser vi?





(he/him)

Alexander Arvidsson

Lead Data Transformation Architect
@ Advania

Utbildare | Talare

advania

KNEE-DEEP IN TECH TOPICS

MVP Microsoft® Most Valuable Professional

Del I

Vad

Komponenterna



Microsoft 365 Copilot



Copilot Studio



Fabric Data Agent



Fabric Operations Agent



Foundry

Copilot och Foundry klarar sig utan Fabric.

Affärsnytta från era siffror gör det inte.

Deterministiskt eller agentiskt

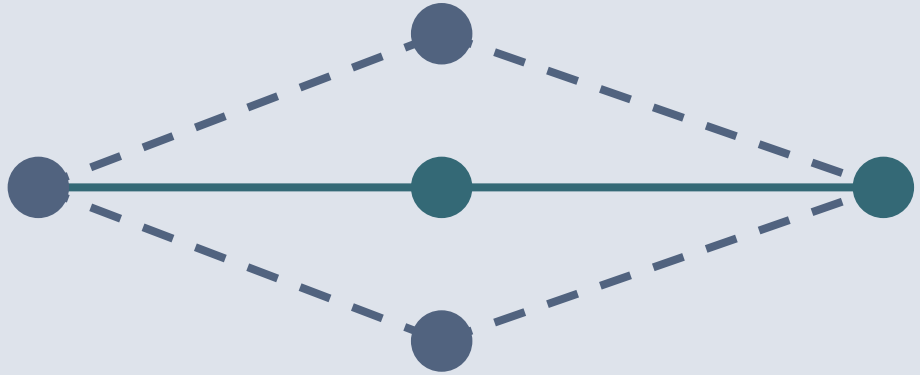
Deterministiskt



Samma väg varje gång

Testa allt

Agentiskt



Modellen väljer väg

Utvärdera ett urval



Copilot Studio: low-code

Byggs av	makers och IT
Körs	som tjänst hos Microsoft
Kostar	Copilot Credits
När	Teams, Copilot och webben



Fabric-agenter: nära datan

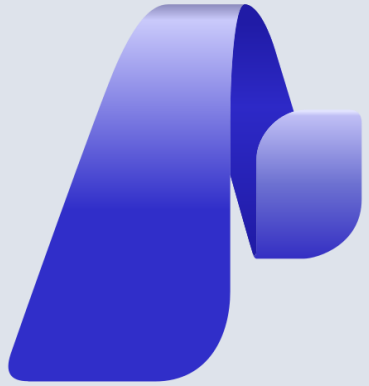
Byggs av	dataägare och BI-team
Körs	i er Fabric-kapacitet
Kostar	kapacitetsenheter (CU)
Når	Studio, Foundry, Teams





Foundry: pro-code

Byggs av	utvecklare
Körs	i din Azure-prenumeration
Kostar	Azure-konsumtion
Ger	modellval, spårning, nätverk



Del II

HUR

Koppla ihop
Gör datan redo
Ta det till drift

En arkitektur, två spår

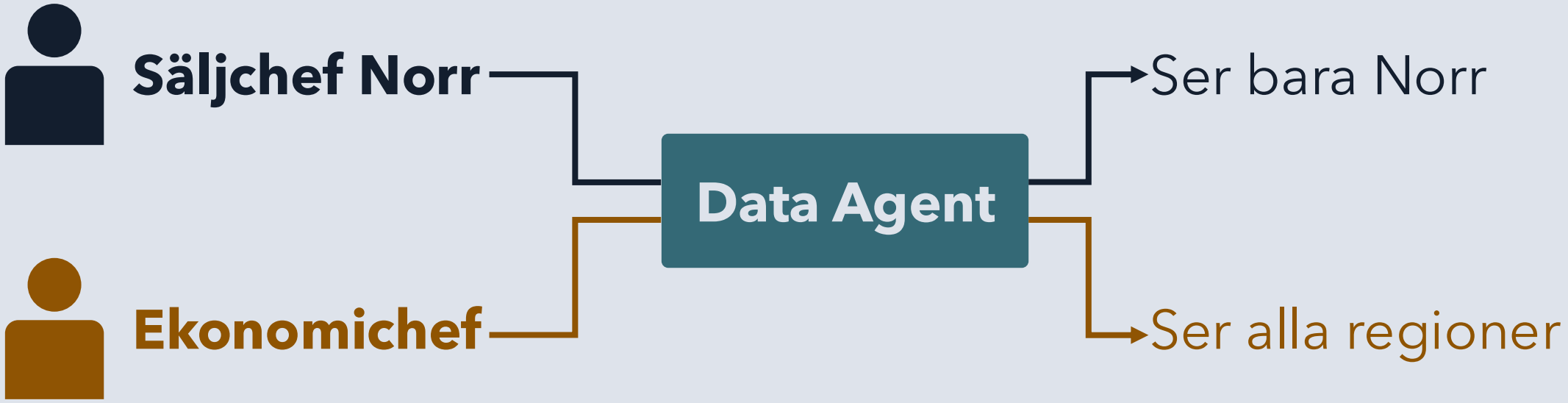


Identitet och styrning hela vägen: Entra Agent ID, Agent 365

Vad når Copilot, och hur?

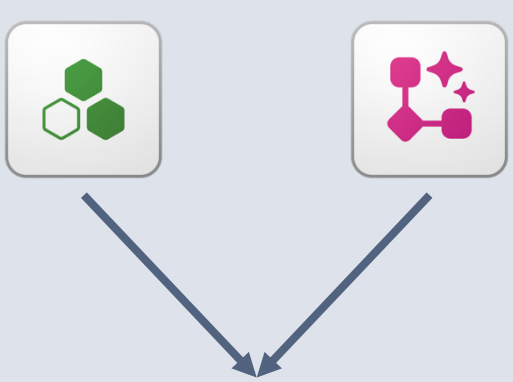
Power BI-modeller	direkt, som sammanhang (context)
Data Agent	kräver en publicerad agent
Ontologi	via Copilot Studio

Frågan körs som användaren



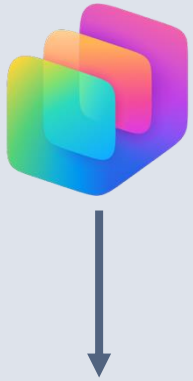
Åtkomst krävs till både agenten och källan.

Kostnaden hamnar olika



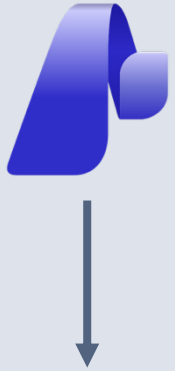
Fabric-kapacitet

fast pris(?)



Copilot Credits

per användning



Azure-konsumtion

per användning

Tre fakturor, ofta hos tre olika team.

Fem begränsningar för Data Agent



Tabeller

inga dokument



Bara läsning

skriver aldrig



Fem källor

per agent



25 rader

per svar



Fast modell

går inte att byta

Den är till för samtal om data, inte för att hämta dataset.



Kvaliteten avgörs i källan.



AI skapar inga nya kvalitetsproblem. Den förstärker dem.

Acceptera Källa → Fabric → **Fel i svaret**

Återför Källa → Fabric → **Rätt svar**



om källan rättas

Modellen är kontraktet med AI

	Före	Efter
Mått	ZZ_AMT_NET	Intäkt, netto
Datum	dt1	Orderdatum
Definition	Saknas	En, i modellen

Ontologin är verksamhetens språk.

Vad saker är

Kund — **Order** — **Produkt**

Hur processerna samverkar

Kund lägger order → **Ordern levereras**

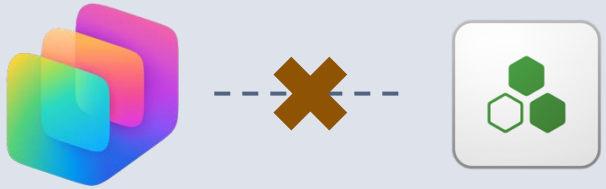
ett åtagande — en intäkt


FLIGHT CONTROLS
TRACKING
PULSE
STOBE
LANDING
RPH
ALT
CIRRUS
PRESS TO TEST

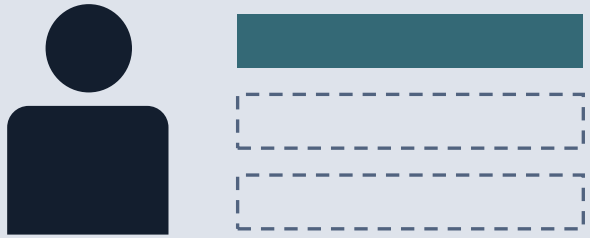
Samma kodbas, olika vägar till drift

Spara		**Git**	
Flytta	Deployment pipelines	Power Platform-pipelines	GitHub Actions, DevOps
Testa	Python-SDK mot facit	Inbyggd utvärdering	Utvärderare i pipeline

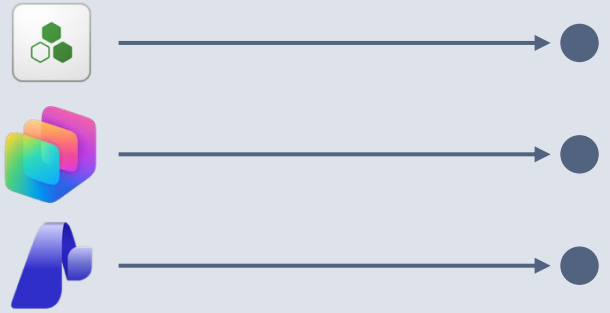
Tre luckor mellan verktygen



Studios test når inte Data Agent



Testet ser bara testarens data



Gemensam release bygger ni själva


T
WX
YZ
U
NEW STANDARD

Vem äger definitionerna under agenten?



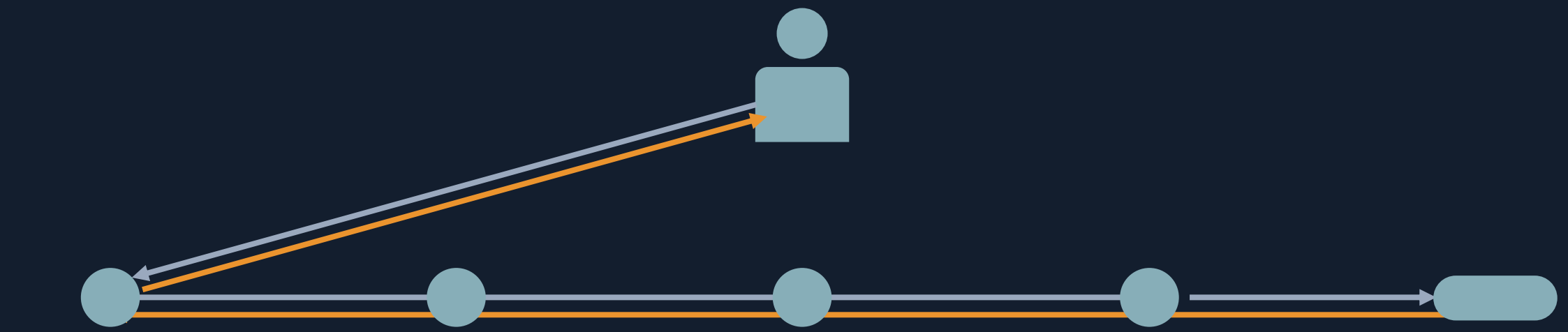
Del III

VARFÖR

Problem och nytta

”Vilka regioner missade Q3-målet?”

En fråga, fyra länkar.



Fem sätt att mäta nyttan

Tid till beslut	före och efter
Avlastning	färre ad hoc-frågor
Användning	återkommande användare
Svarskvalitet	andel rätt mot facit
Kostnad per svar	kapacitet, credits, tokens

Planera för förändring.

Copilot är ytterdörren,
Fabric är grunden

Kedjan blir aldrig bättre än modellen

Nyttan kommer när svaret går att lita på



Förtroende är en färskvara.

STORT TACK!



advania

alexander.arvidsson@advania.se