



Voorstel technische inrichting basisomgeving en data-architectuur Data.OD

EPIC 2: Technisch fundament gerealiseerd

E:projectteam Data.OD

B:Tjiko Schutte

Versie 1.0

Datum mei 2026

EPIC 2: Technisch fundament gerealiseerd.....	1
Voorstel inrichting Azure-tenant & dataplatform Data.OD.....	3
1. Inleiding.....	3
2. Uitgangspunten	3
3. Oplossingsrichtingen	3
3.1 Microsoft Fabric (SaaS)	3
3.2 Azure Database Services (PaaS)	4
4. EasyMorph als bepalende factor.....	4
5. Architectuurvoorstel.....	5
5.1 Raw Layer (Data Lake).....	5
5.2 Performance Layer (Database)	5
5.3 Ontsluiting (Virtueel Datawarehouse)	6
6. Beveiliging en compliance.....	6
7. Overweging andere technologieën	6
8. Toekomstbestendigheid	6
9. Uitgangspunten ingevuld.....	7
10. Conclusie en advies.....	8

Voorstel Inrichting Azure-omgeving Data.OD	10
1. Samenvatting	10
2. Keuze: Dedicated Subscription in ODDV Tenant.....	10
3. Technische Componenten (Infrastructuur)	10
4. Omgevingen: Dev en Prod op één hoop	11
5. Beheer en Kostenmonitoring	11
6. Besluitvorming	11
Bijlage risico's.....	11
1. Beheerafhankelijkheid (Shared Fate).....	11
2. Rechten-vervuiling (Over-privileging).....	12
3. Kosten-discussies	12
4. Continuïteit bij stopzetting samenwerking.....	12
Bijlage Afweging Federatieve Koppeling versus Gastaccounts binnen Azure	13
1. Licentiekosten en afhankelijkheid van premium-functionaliteit.....	13
2. Technische beperkingen van federatieve toegang	13
3. Beheerbaarheid en operationele afhankelijkheden	14
4. Governance, controle en veiligheid	14
Conclusie.....	15
Voorstel inrichting netwerk en toegang.....	16
1. De Voorkeursopties (Architectuurkeuzes).....	16
Identiteit & Authenticatie (De Sleutel):.....	16
Netwerktogang (De Voordeur):	16
2. Toegang tot Data (Specifiek voor de Stuurgroep)	17
3. Consequenties van deze keuzen	18
4. Toegang per rol (RBAC Matrix)	18
Vereenvoudigde schets	18
Onderdelen en kosten	19
Samenvattend	20

Voorstel inrichting Azure-tenant & dataplatform Data.OD

1. Inleiding

Voor de realisatie van Data.OD is het noodzakelijk om een fundamentele keuze te maken in de inrichting van de Azure-tenant en het onderliggende dataplatform. Deze keuze bepaalt in grote mate de ontwikkelsnelheid, beheersbaarheid, kosten en toekomstbestendigheid van de oplossing.

Het doel van dit document is om een concreet en onderbouwd voorstel te doen voor de technische basisomgeving, zodat de gedelegeerde stuurgroep een besluit kan nemen en het ontwikkelteam gericht kan starten met de realisatie.

De centrale vraag binnen dit voorstel is de keuze tussen een geïntegreerde SaaS-oplossing (Microsoft Fabric) of een modulaire PaaS-opzet (Azure Database Services).

2. Uitgangspunten

Bij het bepalen van de juiste oplossingsrichting zijn de volgende uitgangspunten leidend:

- Het platform moet praktisch en betrouwbaar functioneren
- De oplossing moet kostenefficiënt zijn
- Integratie met bestaande tooling (met name EasyMorph) is essentieel
- De architectuur moet toekomstvast zijn, zonder onnodige complexiteit
- Beheer en beveiliging moeten voorspelbaar en beheersbaar blijven
- Het platform moet geschikt zijn voor gebruik door meerdere omgevingsdiensten

3. Oplossingsrichtingen

3.1 Microsoft Fabric (SaaS)

Microsoft Fabric is een geïntegreerde data-oplossing waarin opslag, verwerking en rapportage samenkomen binnen één platform. Het biedt een snelle start doordat infrastructuur grotendeels wordt beheerd door Microsoft.

Voordelen:

- Alles-in-één platform (data engineering, opslag, BI)
- Minimale infrastructuurconfiguratie nodig
- Strategische richting van Microsoft
- Snelle onboarding van nieuwe functionaliteiten

Nadelen:

- Werkt met een capacity-model (vaste kosten, ook bij lage benutting)
- Kans op throttling bij piekbelasting
- Extra abstractielaag (OneLake) → latency en minder controle
- Complexere integratie met externe tools
- Minder geschikt voor samenwerking met EasyMorph
- Beveiligingsmodel (o.a. RLS) nog minder volwassen en in ontwikkeling
- Sterke afhankelijkheid van Microsoft-ecosysteem (vendor lock-in)

3.2 Azure Database Services (PaaS)

Bij deze aanpak wordt het dataplatform opgebouwd uit losse Azure-componenten, zoals Azure SQL en Data Lake Storage. Dit biedt maximale flexibiliteit en controle over de inrichting.

Voordelen:

- Volledige controle over architectuur en configuratie
- Directe en stabiele integratie met EasyMorph
- Gebruik van Azure SQL Serverless → kosten op basis van gebruik
- Geen throttling zoals bij vaste capacity-modellen
- Mature beveiligingsopties zoals Row-Level Security
- Modulair en flexibel uitbreidbaar
- Geen lock-in; eenvoudig door te ontwikkelen richting andere oplossingen

Nadelen:

- Meer initiële inrichting nodig
- Iets meer beheer op componentniveau
- Minder “out-of-the-box” geïntegreerd dan Fabric

4. EasyMorph als bepalende factor

Binnen OddV vormt EasyMorph de kern van de dataverwerking. De keuze voor het dataplatform moet hier direct op aansluiten.

Bij gebruik van Azure SQL (PaaS):

- Directe communicatie met de database-engine
- Hoge performance bij bulkverwerking

- Voorspelbaar gedrag bij gelijktijdige processen
- Eenvoudige en stabiele authenticatie

Bij gebruik van Fabric:

- Communicatie via OneLake en SQL-endpoints
- Extra vertaalslagen naar opslagformaten (Delta/Parquet)
- Grotere kans op latency en vertraging
- Complexere authenticatie (tokens/service principals)
- Risico op throttling bij parallelle dataloads

Conclusie: Voor de huidige werkwijze biedt PaaS een duidelijk stabielere en efficiëntere basis.

5. Architectuurvoorstel

De voorgestelde architectuur bestaat uit twee hoofdlagen:

5.1 Raw Layer (Data Lake)

- Gebruik van Azure Data Lake Storage (ADLS Gen2)
- Opslag van alle brondata in originele vorm (XML, JSON, CSV, etc.)
- Data wordt “as-is” opgeslagen

Voordelen:

- Volledig toekomstvast
- Basis voor verdere analyse en hergebruik
- Direct compatibel met toekomstige platformen (zoals Fabric)

5.2 Performance Layer (Database)

- Gebruik van Azure SQL Serverless
- Data wordt via EasyMorph getransformeerd naar relationele tabellen

Voordelen:

- Hoge performance voor rapportages en queries
- Lage kosten door automatisch schalen
- Optimale aansluiting op bestaande processen

5.3 Ontsluiting (Virtueel Datawarehouse)

- Eén centrale database voor alle omgevingsdiensten
- Scheiding van data via Row-Level Security (RLS)
- Toegang via service principals per organisatie

Dit zorgt voor:

- Eenvoudige beheerstructuur
- Schaalbaarheid naar meerdere partijen
- Veilig gescheiden toegang binnen één omgeving

6. Beveiliging en compliance

De voorgestelde oplossing voldoet aan de benodigde beveiligings- en compliance-eisen:

- Integratie met Microsoft Entra ID
- Ondersteuning voor Multi-Factor Authentication
- Encryptie van data (in transit en at rest)
- Netwerkbeveiliging via IP-restricties en firewallregels
- Gebruik van bewezen technieken (zoals RLS)

7. Overweging andere technologieën

Azure Databricks

Niet gekozen vanwege:

- Hoge complexiteit
- Gericht op data science (Python/Scala)
- Geen meerwaarde voor huidige use-case
- Extra kosten en beheerlast

Azure Synapse

Niet gekozen vanwege:

- Verminderde strategische focus vanuit Microsoft
- Wordt functioneel ingehaald door Fabric
- Risico op toekomstige migratie

8. Toekomstbestendigheid

De voorgestelde PaaS-architectuur is expliciet ontworpen als “Fabric-ready”:

- Gebruik van open opslagformaten (zoals Parquet)
- Data opgeslagen in ADLS (basis van Fabric/OneLake)
- Mogelijkheid om later eenvoudig Fabric “bovenop” te plaatsen

Hierdoor ontstaat:

- Geen lock-in
- Geen dubbele investeringen
- Maximale flexibiliteit richting de toekomst

9. Uitgangspunten ingevuld

Uitgangspunt	Microsoft Fabric (SaaS)	Azure Database Services (PaaS)
Praktisch & betrouwbaar functioneren	Gestandaardiseerde omgeving, snel inzetbaar. Kans op throttling bij piekbelasting en afhankelijk van abstractielagen (OneLake), wat performance minder voorspelbaar maakt.	Bewezen en stabiele componenten. Directe communicatie met database zorgt voor voorspelbare performance en hoge betrouwbaarheid.
Kostenefficiëntie	Capacity-based model met vaste kosten. Minder efficiënt bij wisselend gebruik; capaciteit wordt vaak niet volledig benut.	Pay-per-use model (met name SQL Serverless). Kosten sluiten direct aan op gebruik, zeer efficiënt bij batchverwerking.
Integratie met EasyMorph	Mogelijk, maar indirect via extra lagen. Hogere complexiteit, kans op latency en complexere authenticatie.	Directe en native integratie met Azure SQL. Hoge performance, stabiel en eenvoudig te beheren.
Toekomstvastheid	Strategische richting van Microsoft. Wel nog sterk in ontwikkeling en regelmatig wijzigingen in functionaliteit.	Modulair en gebaseerd op open standaarden (SQL, Parquet). Volledig compatibel met toekomstige uitbreiding richting Fabric.
Complexiteit architectuur	Alles-in-één oplossing, maar introduceert nieuwe concepten en afhankelijkheden. Minder transparant.	Eenvoudige en duidelijke opbouw met losse componenten. Goed te begrijpen en gericht in te richten.
Beheer & beveiliging	Minder infra-beheer, maar complexer securitymodel (combinatie opslag/compute). Minder volwassen in detailcontrole.	Volledig controleerbaar. Bewezen technieken zoals Row-Level Security eenvoudig en betrouwbaar toe te passen.

Multi-tenant gebruik (OD's)	Mogelijk, maar datasegregatie minder volwassen en complexer in inrichting.	Zeer geschikt. Centrale database met RLS maakt veilige scheiding tussen omgevingsdiensten eenvoudig.
Vendor lock-in	Hoog: sterke afhankelijkheid van Fabric-ecosysteem en werkwijze.	Laag: open standaarden en modulaire opbouw voorkomen lock-in.
Performance bij piekbelasting	Beperkt door capacity → throttling mogelijk.	Automatische schaalbaarheid zonder throttling binnen ingestelde grenzen.
Geschiktheid voor huidige werkwijze	Minder passend; stuurt richting andere tools en werkwijzen.	Sluit direct aan op bestaande EasyMorph-gedreven workflow.

10. Conclusie en advies

Op basis van de bovenstaande analyse wordt geadviseerd om Data.OD in te richten op basis van Azure Database Services (PaaS), met:

- Azure SQL Serverless voor gestructureerde data
- Azure Data Lake Storage (ADLS Gen2) voor ongestructureerde data
- Integratie met EasyMorph als primaire verwerkingslaag

Deze oplossing biedt de beste balans tussen:

- Kosten
- Performance
- Beheersbaarheid
- Aansluiting op de huidige werkwijze
- Toekomstbestendigheid

Daarnaast blijft de mogelijkheid bestaan om op een later moment gecontroleerd door te groeien naar Microsoft Fabric, zonder dat huidige investeringen verloren gaan.

Wij adviseren daarom om te kiezen voor een PaaS-gebaseerde inrichting in plaats van Microsoft Fabric.

Dit biedt op dit moment maximale controle over data-opslag, beveiliging en toegangsmodellen (zoals RLS en netwerkrestricties).

Tegelijkertijd blijft de architectuur zodanig opgezet dat toekomstige migratie of integratie met Fabric mogelijk blijft.

Eindadvies: Kies voor een modulaire PaaS-architectuur als fundament voor Data.OD.

Voorstel Inrichting Azure-omgeving Data.OD

1. Samenvatting

Om Data.OD snel en veilig te kunnen starten, is een duidelijke landing in Azure nodig. Dit voorstel adviseert een dedicated Subscription binnen de bestaande ODDV Azure Tenant. Dit garandeert volledige financiële en technische scheiding van bestaande I-GO/ODDV data, terwijl we gebruikmaken van de bestaande beheerorganisatie en security-oplossingen.

2. Keuze: Dedicated Subscription in ODDV Tenant

Zowel DCMR als ODDV hebben al Azure-omgevingen. De meest robuuste keuze is om de Data.OD omgeving onder te brengen bij de beheerorganisatie (ODDV), maar in een volledig eigen "vakje".

- De Tenant (Entra ID): We maken gebruik van de bestaande ODDV-tenant. Dit betekent dat we meeliften op hun Microsoft Entra ID (voorheen Azure AD) voor authenticatie. Beheerders van ODDV en DCMR (via gastaccounts) loggen in met hun eigen organisatie-account. De keuze voor gastaccounts is beschreven in bijlage 1.
- De Subscription: Binnen de tenant wordt een nieuwe, aparte subscription aangemaakt voor Data.OD. Een subscription is de hardste grens in Azure voor beheer en kosten.

Waarom deze opzet?

1. Strikte Isolatie: Beheerders van ODDV of DCMR hebben geen toegang tot de data in Data.OD, tenzij ze expliciet rechten krijgen op deze specifieke subscription.
2. 100% Kostentransparantie: Er ontstaat geen discussie over facturen. De kosten die op deze subscription vallen, zijn 1:1 toe te wijzen aan Data.OD.
3. Beheergemak: De ODDV voert de regie op het platform, maar DCMR-beheerders kunnen eenvoudig ondersteuning bieden via gedelegeerde toegang (RBAC) op de resources.

3. Technische Componenten (Infrastructuur)

Om het platform te laten draaien conform onze werkwijze (praktisch en doelgericht), richten we de volgende basiscomponenten in binnen de subscription:

- ETL / Batch Server: Een dedicated VM (Windows Server) voor de EasyMorph-motor. Dit is het werkpaard voor de inlaad- en transformatieprocessen.
- Toegang (Gateway): Om veilig bij de servers en databases te komen zonder deze open te zetten op het publieke internet, richten we een Azure Bastion of een Basic VPN Gateway in. Dit is de beveiligde "voordeur" voor beheerders.

- Opslag & Database: Afhankelijk van het eerdere besluit (SaaS vs PaaS) landen hier de SQL Serverless databases en de Blob Storage (ADLS Gen2) voor de Raw Layer of de fabric configuratie.
- Authenticatie: We maken gebruik van Microsoft Entra ID (Free) voor service principals en gebruikerstoegang. Dit zorgt voor veilige koppelingen tussen EasyMorph, Azure SQL en de externe bronnen.

4. Omgevingen: Dev en Prod op één hoop

In afwijking van academische standaarden (OTAP), kiezen we voor snelheid en kostenbeperking. We maken geen aparte subscriptions of resource groups aan voor Dev/Test en Productie.

- We maken gebruik van deels dezelfde resources (bijv. één SQL server met verschillende databases).
- Scheiding vindt plaats op logisch niveau (naamgeving) en via rechtenstructuren.

5. Beheer en Kostenmonitoring

- Landing bij ODDV: De omgeving wordt technisch beheerd door ODDV.
- Co-beheer: DCMR-beheerders kunnen op afroep bijspringen via specifieke RBAC-rollen (Contributor op resource-niveau).
- Dashboarding: Optioneel kan er Azure Cost Management dashboard worden ingericht met wekelijkse rapportages naar de stuurgroep. Kosten per databron worden inzichtelijk gemaakt door middel van Tagging op de resources. Dit voor een latere fase.

6. Besluitvorming

Aan de stuurgroep wordt gevraagd in te stemmen met:

1. Het aanmaken van een nieuwe Azure Subscription binnen de ODDV tenant.
2. Het toekennen van beheerrollen aan de ODDV (regie) en DCMR (ondersteuning).
3. De inrichting van de ETL-server en gateway voor directe realisatie van de MVP.

Bijlage risico's

Als we de Data.OD omgeving als een eigen subscription binnen de ODDV-tenant laten landen, zijn de risico's technisch en organisatorisch goed te overzien. Het is een pragmatische route die snel resultaat geeft zonder de boel onveilig te maken.

Dit zijn de specifieke risico's rekening mee gehouden moet worden:

1. Beheerafhankelijkheid (Shared Fate)

Omdat we in de tenant van ODDV zitten, liften we mee op hun centrale instellingen.

- **Risico:** Als de centrale IT van ODDV een fout maakt in de globale security-politicijs (bijvoorbeeld een te strikte Conditional Access policy), kan onze ETL-server of de toegang voor DCMR-collega's onbedoeld platvallen.
- **Mitigatie:** We leggen bij de start harde afspraken vast over de "beheergrens": Data.OD wordt ingericht als een technisch geïsoleerde subscription. Hierdoor borgen we dat globale tenant-scripts of generieke wijzigingen vanuit de centrale ODDV-omgeving onze specifieke resources niet onbedoeld raken of verstoren.

2. Rechten-vervuiling (Over-privileging)

Binnen een tenant is het verleidelijk om mensen 'Global Admin' te maken of te ruime rechten te geven op tenant-niveau.

- **Risico:** Een beheerder die eigenlijk alleen bij de I-GO omgeving van ODDV moet zijn, krijgt per ongeluk ook rechten op de Data.OD subscription.
- **Mitigatie:** We hanteren strikte RBAC-rollen op subscription-niveau. Dit houdt in dat niemand rechten krijgt binnen de Data.OD-omgeving, tenzij zij direct deel uitmaken van het project- of beheerteam. Hiermee borgen we dat de toegang tot de landelijke voorziening volledig gescheiden blijft van de reguliere ODDV-beheerders.

3. Kosten-discussies

Hoewel we een eigen subscription hebben, komt er één verzamelfactuur bij de ODDV binnen.

- **Risico:** Discussie over de verdeling van de factuur of onduidelijkheid over welke resources de kosten drijven.
- **Mitigatie:** We passen consequent Azure Tags toe op alle resources binnen de subscription. Door elke component (zoals de ETL-server, SQL-database en Gateway) te labelen met Project: Data.OD, borgen we dat de volledige Azure-factuur met één druk op de knop kan worden gesplitst. Dit elimineert elk financieel risico op onduidelijkheid over de kostenverdeling tussen de deelnemende diensten.

4. Continuïteit bij stopzetting samenwerking

- **Risico:** Als de samenwerking tussen ODDV en de rest van het netwerk stopt, "gijzelt" de ODDV-tenant de omgeving.
- **Mitigatie:** We richten de volledige infrastructuur in binnen een eigen, specifieke subscription. Hierdoor borgen we de onafhankelijkheid van de landelijke voorziening: mocht de samenwerking met ODDV wijzigen of de regie overgaan

naar een andere partij (zoals DCMR of een nieuwe stichting), dan is een Subscription Move naar een andere tenant een relatief eenvoudige technische handeling die zonder downtime kan worden uitgevoerd.

Conclusie: Het risico is laag, mits we de grens van de subscription bewaken met RBAC en de data binnenin filteren met RLS. Het is technisch een stuk veiliger dan alles in één bestaande subscription "erbij" proppen.

Bijlage Afweging Federatieve Koppeling versus Gastaccounts binnen Azure

1. Licentiekosten en afhankelijkheid van premium-functionality

De huidige inrichting maakt gebruik van de gratis functionaliteit binnen Microsoft Entra ID. Bij een federatieve inrichting via B2B Direct Connect ontstaat echter direct een afhankelijkheid van premium-licenties.

Voor het configureren en beheren van Cross-tenant access settings — de basis van federatieve toegang — vereist Microsoft Entra ID P1- of P2-licenties aan beide zijden van de koppeling. Dit betekent dat zowel ODDV als DCMR (en toekomstige deelnemende omgevingsdiensten) over deze licenties moeten beschikken.

Hiermee ontstaat een structurele afhankelijkheid van aanvullende Microsoft-licenties, terwijl dit functioneel niet noodzakelijk is voor het beoogde gebruik van data.OD.

Gastaccounts (B2B Collaboration) kennen deze beperking niet. Microsoft biedt hiervoor een ruim gratis model op basis van Monthly Active Users (MAU), waarmee de huidige en verwachte omvang van het samenwerkingsverband ruimschoots ondersteund kan worden zonder aanvullende licentiekosten.

2. Technische beperkingen van federatieve toegang

B2B Direct Connect is door Microsoft primair ontworpen voor samenwerking binnen Microsoft Teams Shared Channels (Teams Connect). De functionaliteit is daarmee beperkt in toepassingsgebied.

Voor beheer en gebruik van Azure-resources — zoals:

- Azure Portal toegang;
- toegang tot Synapse- of SQL-omgevingen;
- beheer van storage- en dataresources;

biedt federatieve toegang via Direct Connect momenteel geen volwaardige ondersteuning.

In de praktijk betekent dit dat, wanneer externe beheerders of ontwikkelaars daadwerkelijk binnen de Azure-omgeving van ODDV moeten kunnen werken, alsnog gebruik gemaakt moet worden van gastaccounts (B2B Collaboration).

Een federatieve inrichting lost daarmee het functionele vraagstuk rondom beheer van de Azure-infrastructuur niet op en introduceert tegelijkertijd extra complexiteit.

3. Beheerbaarheid en operationele afhankelijkheden

Een federatieve koppeling vereist wederzijdse configuratie en afstemming tussen beide organisaties. Zowel inbound als outbound policies moeten correct ingericht en beheerd worden.

Dit betekent concreet:

- wijzigingen aan security policies bij één partij kunnen de samenwerking verstoren;
- troubleshooting wordt afhankelijk van meerdere beheerorganisaties;
- wijzigingen moeten onderling afgestemd worden;
- beheerprocessen worden complexer en minder voorspelbaar.

Voor een samenwerking tussen twee organisaties is dit al een aandachtspunt. Binnen de ambitie van data.OD — waarbij potentieel meerdere omgevingsdiensten aansluiten — schaalt deze complexiteit aanzienlijk op.

Bij een federatief model moeten afzonderlijke koppelingen en afspraken onderhouden worden met iedere deelnemende organisatie. Dit leidt tot:

- hogere beheerlast;
- langere doorlooptijden bij onboarding;
- grotere afhankelijkheid van externe IT-afdelingen;
- verhoogde kans op verstoringen.

Gastaccounts zijn vanuit beheerperspectief aanzienlijk pragmatischer:

- gebruikers kunnen direct worden uitgenodigd;
- rechten worden centraal beheerd binnen de ODDV-tenant;
- onboarding van nieuwe deelnemers is eenvoudig;
- wijzigingen blijven beheersbaar binnen één centrale omgeving.

4. Governance, controle en veiligheid

Bij gebruik van gastaccounts blijft de volledige autorisatie en governance centraal beheerd binnen de ODDV-tenant. Hierdoor:

- blijft controle over toegang volledig bij de eigenaar van het platform;
- kunnen rollen en rechten eenduidig worden toegewezen;

- kunnen accounts eenvoudig worden ingetrokken of aangepast;
- blijft auditing centraal beschikbaar.

Dit sluit beter aan bij een pragmatische en beheersbare inrichting van een gedeeld dataplatform.

Conclusie

Hoewel federatieve koppelingen aantrekkelijk kunnen lijken vanuit het idee van “single sign-on” tussen organisaties, blijkt deze inrichting in de praktijk minder passend voor het beheren van Azure-dataresources binnen een gezamenlijke dataomgeving.

Een federatieve inrichting:

- introduceert afhankelijkheid van premium-licenties;
- biedt beperkte ondersteuning voor Azure-resourcebeheer;
- verhoogt de technische en organisatorische complexiteit;
- schaalt beperkt bij groei naar meerdere omgevingsdiensten.

Gastaccounts binnen de ODDV-tenant bieden daarentegen:

- een direct werkbare oplossing;
- lagere kosten;
- centrale governance;
- eenvoudiger beheer;
- betere schaalbaarheid richting toekomstige deelnemers.

Op basis van bovenstaande overwegingen heeft een inrichting op basis van Azure B2B gastaccounts daarom de duidelijke voorkeur voor de data.OD-omgeving.

Voorstel inrichting netwerk en toegang

Dit voorstel schetst de inrichting van de veilige "voordeur" en de interne rechtenstructuur van het platform.

1. De Voorkeursopties (Architectuurkeuzes)

We kiezen voor een gelaagde aanpak waarbij we gebruikmaken van de bestaande ODDV-tenant, maar met een harde knip op subscription-niveau.

Identiteit & Authenticatie (De Sleutel):

- Microsoft Entra ID: We liften mee op de tenant van ODDV voor het identiteitsbeheer. Voor alle menselijke gebruikers (zowel intern als via B2B gastaccounts) is MFA (Multifactor Authentication) verplicht. Hiermee borgen we dat enkel een gebruikersnaam/wachtwoord combinatie nooit voldoende is om toegang te krijgen tot de SQL-database of andere resources."
- B2B Collaboration: Geen gedoe met extra wachtwoorden; medewerkers van DCMR en andere OD's loggen in via gastaccounts met hun eigen organisatie-account, dit door middel van multifactor authenticatie.
- Service Principals: Voor alle "machine-to-machine" processen (zoals EasyMorph die data uit de Raw Layer haalt of naar SQL schrijft) gebruiken we Service Principals. Dit borgt continuïteit als er mensen uit het projectteam vertrekken. *Voor externe systemen en geautomatiseerde processen maken we gebruik van Service Principals met certificaten of secrets. Dit elimineert de noodzaak voor persoonlijke SQL-accounts en zorgt ervoor dat toegangsrechten fijnmazig (RBAC) beheerd kunnen worden via Entra ID, zonder dat er statische wachtwoorden in de applicatieconfiguratie blijven staan.*

Netwerktoegang (De Voordeur):

De toegang tot de Azure SQL Database is gebaseerd op het 'Zero Trust' principe. Dit betekent dat whitelisting op IP-niveau slechts de eerste barrière is. De daadwerkelijke toegang wordt pas verleend op basis van de identiteit (via de Service Principal of MFA-geauthenticeerde gebruiker). Zelfs als een IP-adres vervalst zou worden, is zonder de juiste Entra ID-token geen toegang tot de data mogelijk.

De volgende onderdelen worden daarbij geïmplementeerd:

- Azure Application Gateway (WAF): Voor de ontsluiting van API's en webverkeer. De buitenwereld praat tegen de Gateway op poort 443. De Web Application Firewall (WAF) inspecteert het verkeer op dreigingen en stuurt het intern door naar de ETL-server. Zo blijft de server zelf onzichtbaar voor het publieke internet.

De ETL-server fungeert als centrale verwerkingscomponent binnen het platform. Deze keuze is bewust gemaakt vanuit eenvoud, voorspelbaarheid en beheersbaarheid.

Schaalbaarheid wordt in deze fase primair verticaal opgelost (opschalen VM). Bij toenemende belasting of beschikbaarheidseisen kan deze component in een latere fase worden opgesplitst of vervangen door gedistribueerde PaaS-diensten.

Hiermee is dit geen beperkende architectuurkeuze, maar een gefaseerde implementatie.

- Azure NAT Gateway: Voor het bevragen van externe bronnen (SFTP, Mail, API's). De server krijgt geen publiek IP, maar gaat naar buiten via één vast uitgaand IP-adres dat door bronhouders gewhitelist kan worden.
- Azure Bastion: Puur voor technisch beheer. Dit is de beveiligde ingang voor beheerders om via de browser op de ETL-server te werken zonder dat RDP-poorten openstaan op het internet.
- Firewall & IP-restricties: De Azure SQL database en Storage Accounts zijn volledig afgeschermd. Toegang tot Azure SQL en Storage wordt in deze fase beperkt via IP-restricties (ETL-server en kantoorlocaties).

Deze aanpak biedt een duidelijke en beheersbare beveiligingslaag met minimale complexiteit.

In een volgende volwassenheidsstap kan dit worden doorontwikkeld naar Private Endpoints en volledige netwerkisolatie (zero-trust), zonder impact op de huidige architectuur. Alleen verkeer van de ETL-server (IP-gebaseerd) en de kantoren van de deelnemende OD's wordt toegelaten.

2. Toegang tot Data (Specifiek voor de Stuurgroep)

De crux voor de stuurgroep zit in de scheiding van data: hoe borgen we dat OD's alleen zien wat voor hen bestemd is?

- Row-Level Security (RLS): In de Azure SQL "Performance Layer" passen we RLS toe. Dit betekent dat er één tabel is met bijvoorbeeld de Externe datasets DCMR aangevuld IGO, maar dat een gebruiker van OD De Vallei via een filter alleen de rijen ziet die relevant zijn voor zijn regio.
 - Row-Level Security (RLS) wordt toegepast op basis van gebruikersidentiteit vanuit Microsoft Entra ID.
 - Indien nodig wordt Object-Level Security toegepast op basis van gebruikersidentiteit en Microsoft Entra ID.
 - Toegang wordt niet individueel per gebruiker beheerd, maar via groepen en/of attributen (bijv. per omgevingsdienst/regio).

- Hiermee blijft de oplossing schaalbaar en beheersbaar bij groei naar meerdere organisaties.
- Data Lake Segregatie: In de "Raw Layer" (ADLS Gen2) werken we met een mappenstructuur waarbij rechten per map worden toegekend via Access Control Lists (ACL's).

3. Consequenties van deze keuzen

Dit zijn de belangrijkste afwegingen:

Keuze	Consequentie / Voordeel
Dedicated Subscription	Hardste grens: 100% isolatie van data en kosten. Fouten in de ODDV-tenant raken Data.OD niet direct.
PaaS boven SaaS (Fabric)	Directe controle: We kunnen fijnmazige beveiliging (zoals RLS) nu al volwassen inzetten, wat in Fabric nog in ontwikkeling is.
Geen OTAP-scheiding*	Pragmatisch & Goedkoop: Lager in kosten en sneller in beheer, maar vraagt om discipline in naamgeving en rechtenbeheer op logisch niveau.
Shared Tenant (ODDV)	Beheergemak: We liften mee op de centrale security-oplossingen en Microsoft Entra ID van ODDV, maar zijn wel afhankelijk van hun globale policies.

* Er is bewust gekozen om in deze fase geen volledige OTAP-straat in te richten. Dit verlaagt kosten en beheercomplexiteit en sluit aan bij de huidige omvang en ontwikkelsnelheid van het platform. Risico's worden gemitigeerd door: Strikte scheiding in naamgeving en rechten (logische scheiding), Beperkte toegang tot wijzigingsrechten, Beheerst releaseproces. Naarmate het platform groeit in gebruik en kritikaliteit kan OTAP gefaseerd worden toegevoegd.

4. Toegang per rol (RBAC Matrix)

Wie	Toegang tot...	Methode
(Architect/Dev)	Volledige Azure Subscription	Role Based Access Control (RBAC) - Contributor
Beheerder (ODDV/DCMR)	Infra, VM en Databases	RBAC en Azure Bastion
EasyMorph (Systeem)	Storage (Raw) en SQL (Performance)	Service Principals
Eindgebruiker (Analist)	Dashboard / SQL Views	Entra ID + Row-Level Security (RLS)

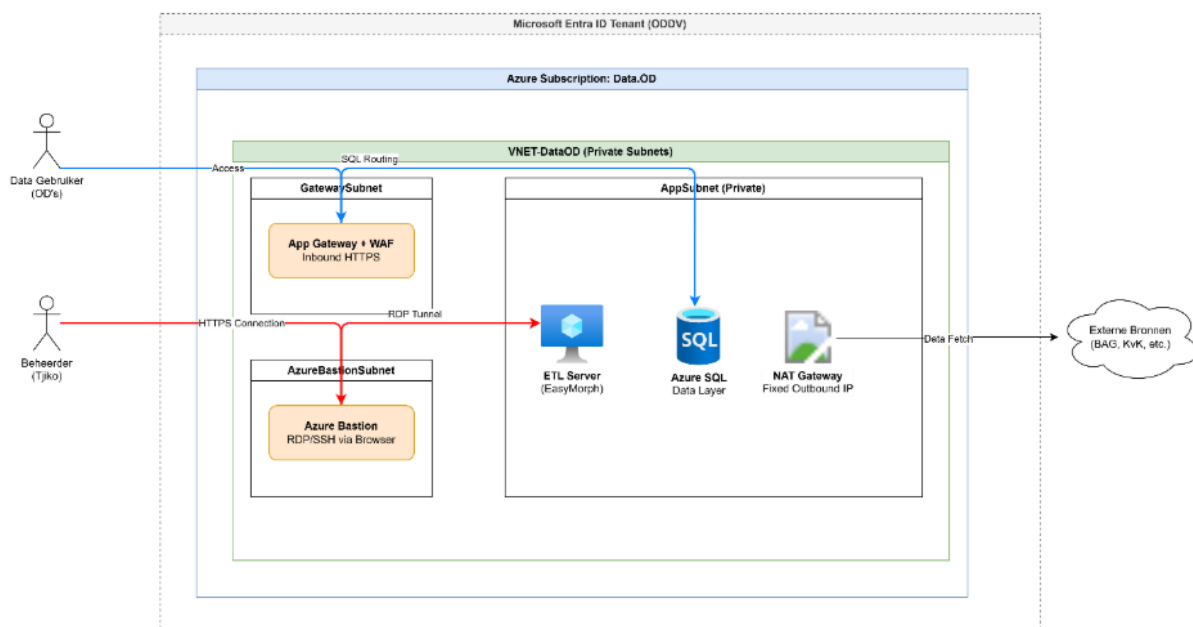
Vereenvoudigde schets

De onderstaande schets laat de componenten zien.

De WAF zorgt er voor dat gebruikers altijd via een beveiligde verbinding data ophalen (data base of eventueel de API ETL server).

Bastion zorgt er voor dat er via een beveiligde manier een connectie gemaakt kan worden via de ETL server.

De NAT Gateway zorgt ervoor dat er data opgehaald kan worden bij externe bronnen zonder er een IP ontsloten wordt naar buiten.



Onderdelen en kosten

De onderstaande tabel laat de kosten en componenten zien:

Component	Service / SKU	Kosten p/m (schatting)	Toelichting
Compute	Standard E4ads v5	€ 345,-	32GB RAM ETL-server (EasyMorph)
Storage (Lake)	Azure Data Lake Storage Gen2 (Hot, LRS)	€ 20,-	Opslag ruwe brondata (100-500GB) incl. transacties
Storage	Premium SSD 150GB	€ 42,-	OS & Data disk voor de VM
Database	SQL Elastic Pool (200 DTU)	€ 410,-	Ruimte voor 100 DB's (50GB incl.)
Beheer	Azure Bastion Standard	€ 280,-	Kan de helft maar dan hebben we geen File Transfer support
Uitgaand	NAT Gateway	€ 35,-	Vast IP voor whitelisting bij bronnen
Inkomend	App Gateway + WAF v2	€ 177,-	De beveiligde voordeur voor OD-gebruikers
Netwerk	VNet, IP & Data	€ 10,-	Basis connectiviteit

Monitoring	Log Analytics / Monitor	€ 10,-	Pay-as-you-go op basis van verbruik
Backup	Recovery Services Vault	€ 10,-	Server & DB backups

De genoemde kosten zijn gebaseerd op huidige inschattingen en gebruik.

Variabele kostencomponenten zoals dataverkeer en logging (Log Analytics) zijn afhankelijk van gebruik en kunnen toenemen bij opschaling.

Monitoring en kostenbewaking worden ingericht om hier tijdig op te kunnen sturen.

Samenvattend

Door te kiezen voor een eigen subscription binnen de ODDV-tenant creëren we een veilige, financieel transparante omgeving die schaalbaar is naar alle 26 omgevingsdiensten, zonder dat we nu al de hoofdprijs betalen voor complexe netwerkinfrastructuren.

De gekozen architectuur is nadrukkelijk ontworpen als een pragmatische, schaalbare basis.

Waar mogelijk is gekozen voor eenvoudige en beheersbare oplossingen, met duidelijke groeipaden richting meer geavanceerde inrichting (zoals zero-trust networking, OTAP en verdere ontkoppeling van componenten).

Hiermee wordt een balans bereikt tussen snelheid, kosten en toekomstvastheid.