



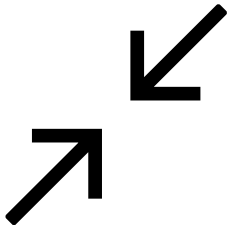
# Cloud Adoption Framework

## For Data Management and Analytics

Amy Genender-Feltheimer

Principal Cloud Solution Architect – Azure Data & AI | FSI Insurance

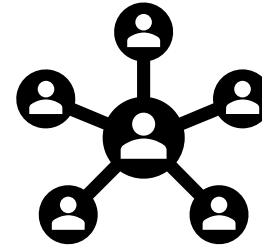
# Agenda



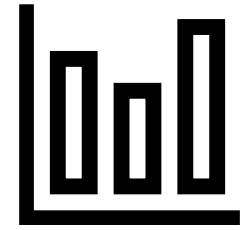
Introduction to CAF  
and WAF



Why we need Data  
Landing Zones




The Data Mesh  
Architecture Paradigm



The CAF for Data  
Management and  
Analytics

# Cloud Adoption Best Practices Progression



Documentation	Description
<a href="#">Cloud Adoption Framework</a>	Managing and governing data is a lifecycle process that starts by building on your existing cloud strategy and carries all the way through to ongoing operations management. The Cloud Adoption Framework will help to guide the full lifecycle for your data estate.
<a href="#">Azure Well-Architected Framework</a>	Workload architecture and operations have a direct impact on data. Understand how your architecture can improve management and governance of the data used by your workload.
<a href="#">Azure landing zones</a>	Choosing a landing zone as the environment to host your workloads, preprovisioned through code establishes a code-based starting point for your Azure environment. Ensure the environment is prepared for the cloud adoption plan through Azure landing zones by starting with Cloud Adoption Framework <a href="#">enterprise-scale landing zones</a> to create the infrastructure of your data platform.
<a href="#">Data management and analytics scenario</a>	Representing a strategic design path for an Azure data management and analytics. The Data management and analytics scenario pattern relies upon distribution of the data and its pipelines across domains. This pattern includes the capabilities such as storage, data lineage, data classification, data ingestion, networking, security, access management, encryption, resiliency and monitoring.

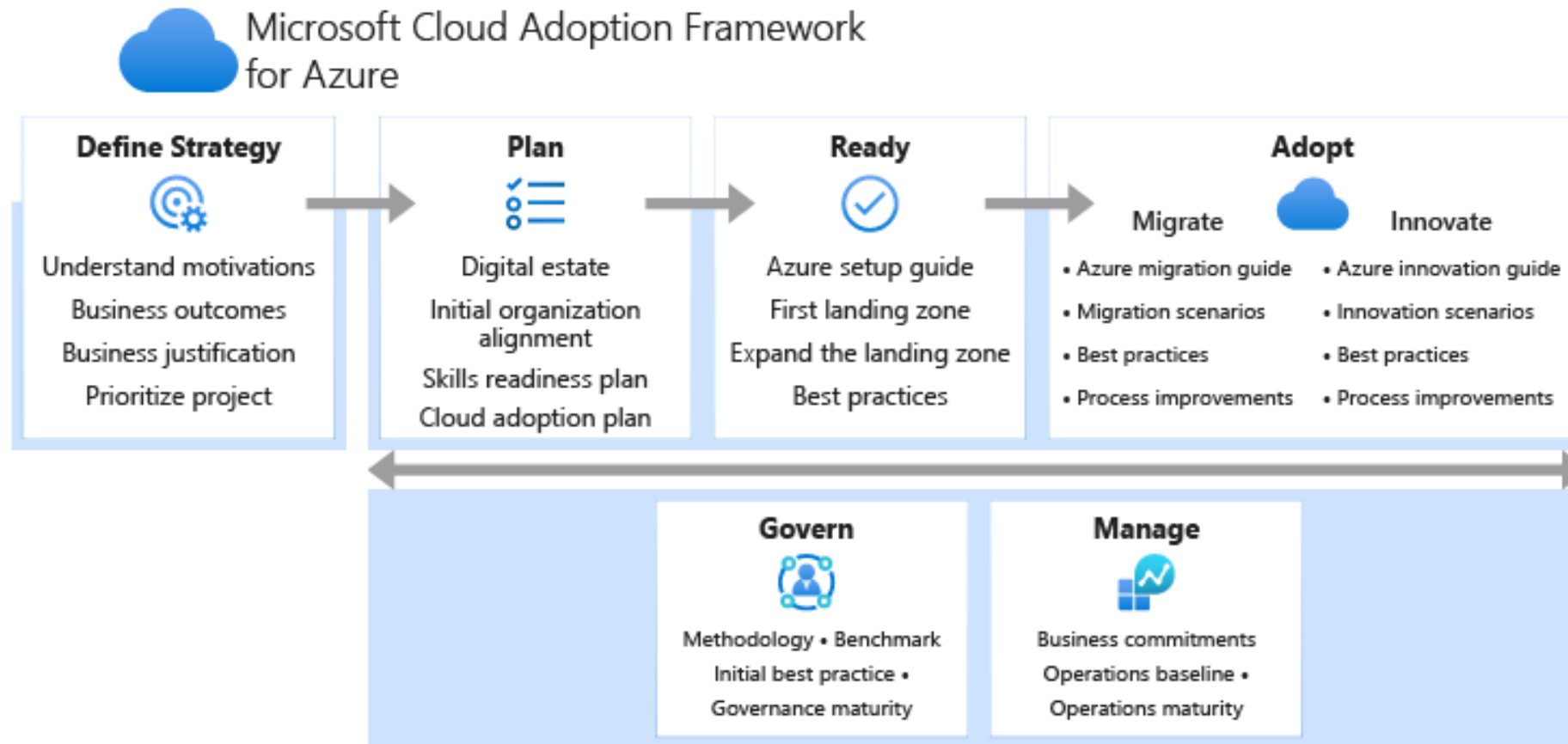


# Cloud Adoption Framework

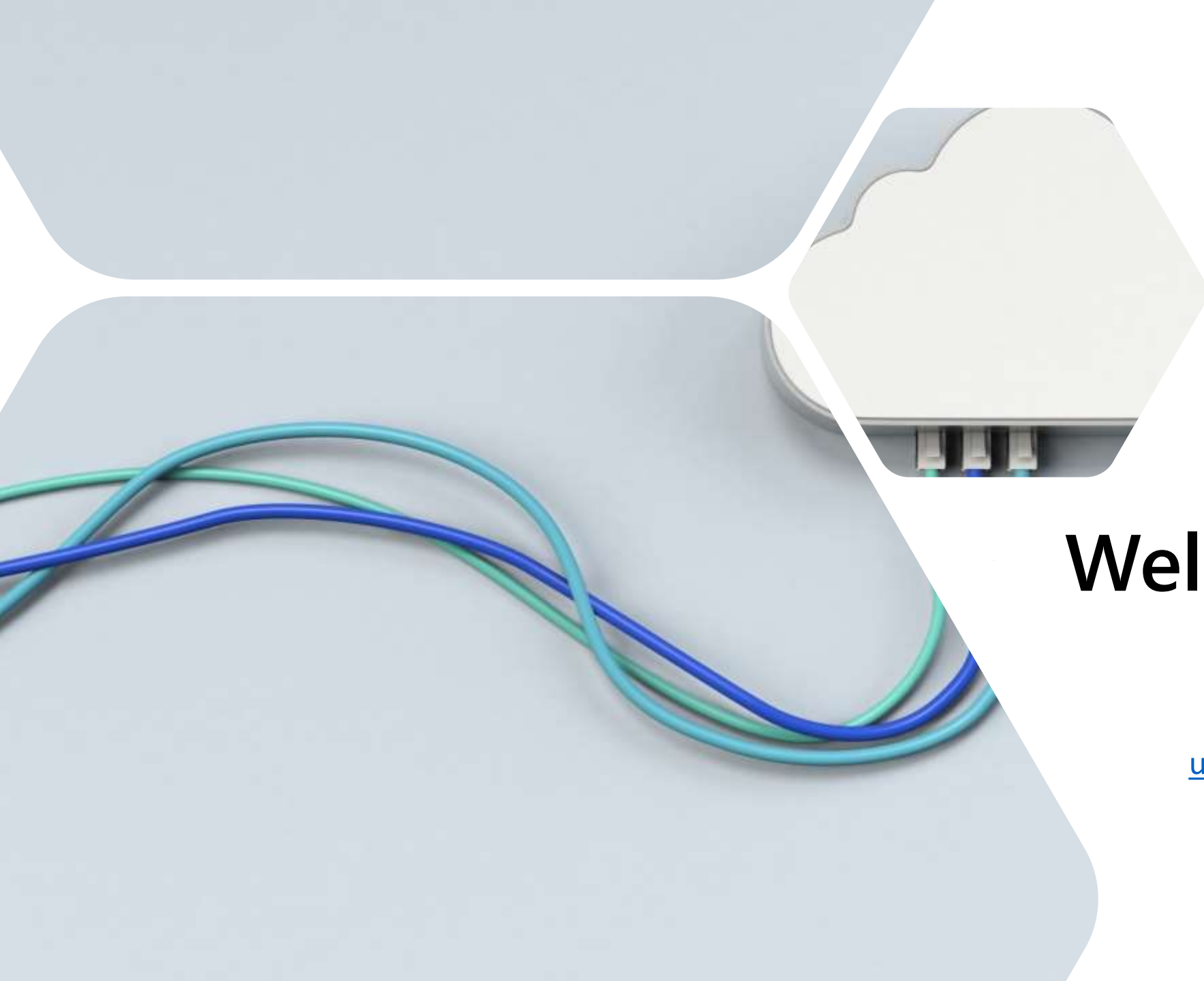
<https://docs.microsoft.com/en-us/azure/cloud-adoption-framework>

# Cloud Adoption Framework

Proven Guidance to Accelerate Your Cloud Adoption Journey



<https://docs.microsoft.com/en-us/azure/cloud-adoption-framework>



# Well Architected Framework

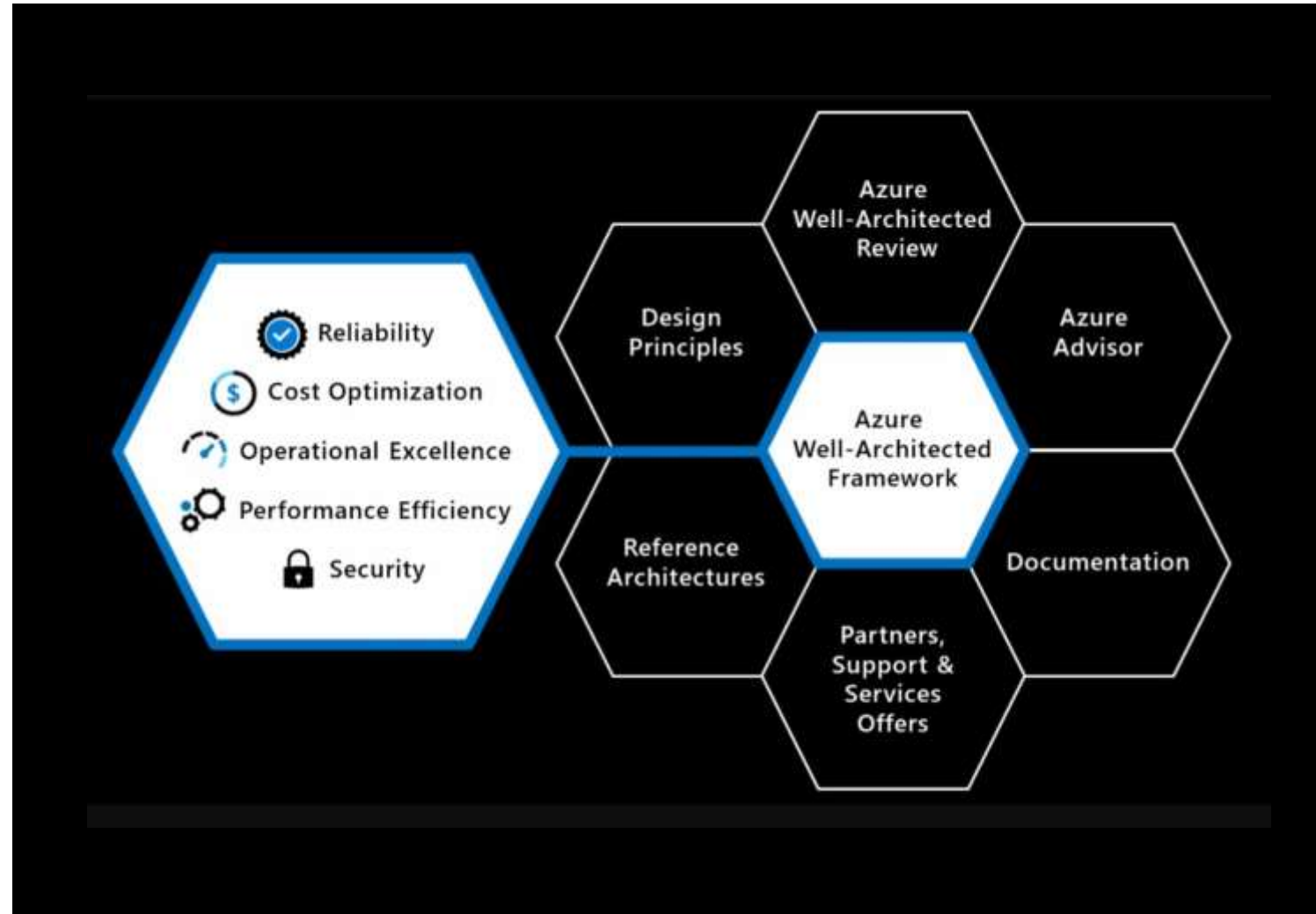
<https://docs.microsoft.com/en-us/azure/architecture/framework>

# Well Architected Framework

The Azure Well-Architected Framework is a set of guiding tenets that can be used to improve the quality of a workload. The framework consists of five pillars of architectural excellence:

- [Reliability](#): The ability of a system to recover from failures and continue to function.
- [Security](#): Protecting applications and data from threats.
- [Cost Optimization](#): Managing costs to maximize the value delivered.
- [Operational Excellence](#): Operations processes that keep a system running in production.
- [Performance Efficiency](#): The ability of a system to adapt to changes in load.

Incorporating these pillars helps produce a high quality, stable, and efficient cloud architecture.





Filter by title

- Data management and analytics Scenario
- Overview
- Build an initial strategy
- Define your plan
- Build data management and analytics capabilities
- > Prepare your environment
- > Govern your data estate
- > Secure your data estate
- > Organize people and teams
- Well-architected considerations
- > Architectures
- > Best practices
- > Featured Azure products



# Implementation guidance

The data management and analytics scenario is designed to guide the end-to-end customer journey through the cloud adoption lifecycle. The full journey requires the following key components or guidance sets:

Documentation	Description
<a href="#">Cloud Adoption Framework</a>	Managing and governing data is a lifecycle process that starts by building on your existing cloud strategy and carries all the way through to ongoing operations management. The Cloud Adoption Framework will help to guide the full lifecycle for your data estate.
<a href="#">Azure Well-Architected Framework</a>	Workload architecture and operations have a direct impact on data. Understand how your architecture can improve management and governance of the data used by your workload.
<a href="#">Azure landing zones</a>	Choosing a landing zone as the environment to host your workloads, preprovisioned through code establishes a code-based starting point for your Azure environment. Ensure the environment is prepared for the cloud adoption plan through Azure landing zones by starting with Cloud Adoption Framework <a href="#">enterprise-scale landing zones</a> to create the infrastructure of your data platform.
<a href="#">Data management and analytics scenario</a>	Representing a strategic design path for an Azure data management and analytics. The Data management and analytics scenario pattern relies upon distribution of the data and its pipelines across domains. This pattern includes the capabilities such as storage, data lineage, data classification, data ingestion, networking, security, access management, encryption, resiliency and monitoring.

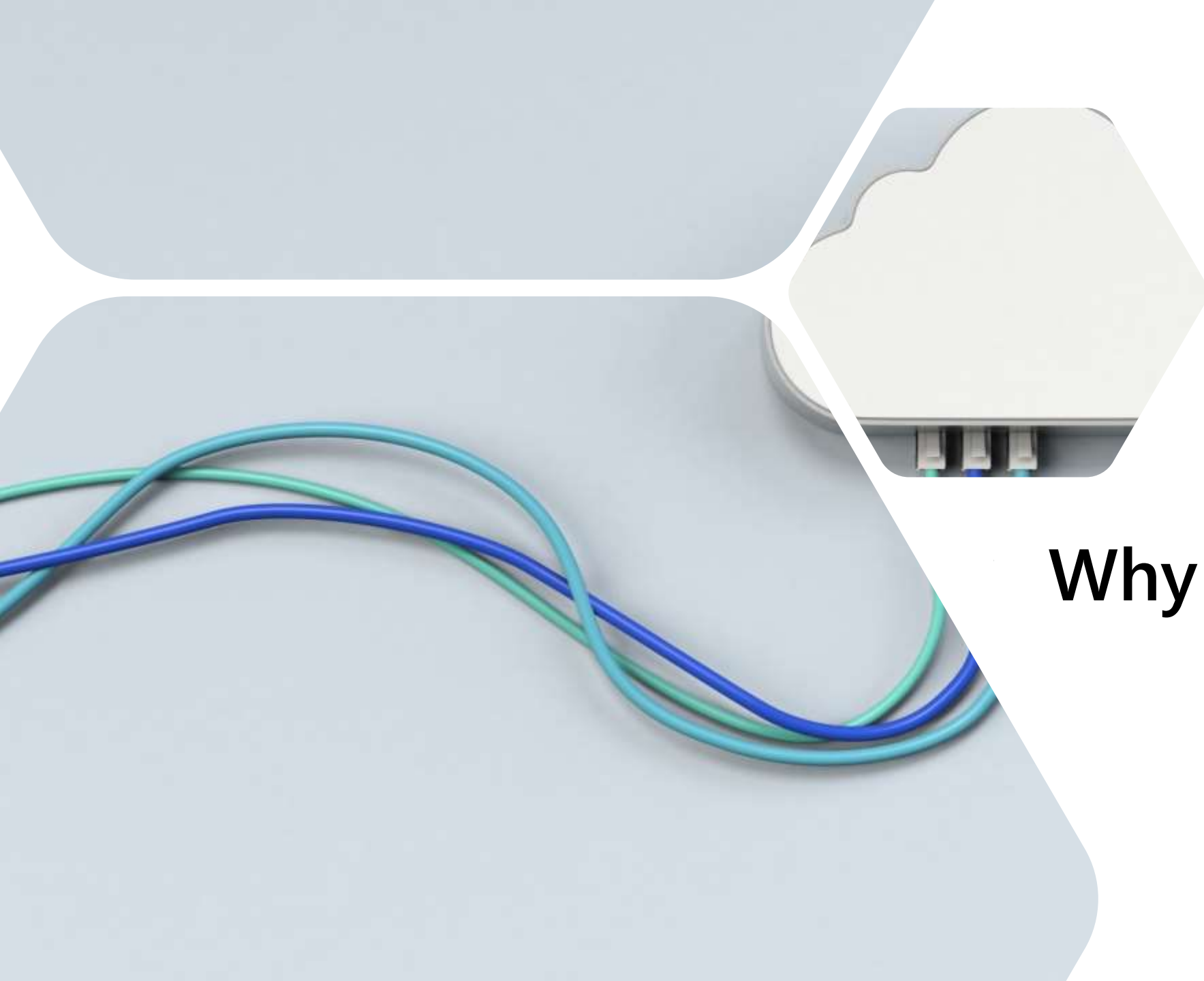
## Tip

Deployable reference implementations and templates make it easy to get started with the data management and analytics scenario. To learn more, see [Deployment templates](#).

## In this article

- [Objectives](#)
- [Implementation guidance](#)
- [Reference architecture](#)
- [Best practices](#)
- [Featured Azure products](#)
- [Common customer journeys](#)
- [Take action](#)
- [Next steps](#)





**Why we need data  
landing zones**

## Azure landing zones

Azure landing zones help customers **set up their Azure environment** for scale, security, governance, networking, and identity.

Azure landing zones:

- Enable **migrations and net new apps**
- Consider **all platform resources**
- **Don't differentiate between IaaS or PaaS**

# The value of creating cloud-ready environments

- ☑ Aligned to business priorities
- ☑ Cloud-design considerations
- ☑ Adapted for cloud operating model
- ☑ Ready for cloud applications
- ☑ Adaptable to grow and expand
- ☑ Compliant



Agile



Cutting-edge innovation



Secure

# Why is this important?

Governance • Compliance • Risk Management



## Citigroup fined \$400 million for risk management



By Clare Duffy, CNN Business

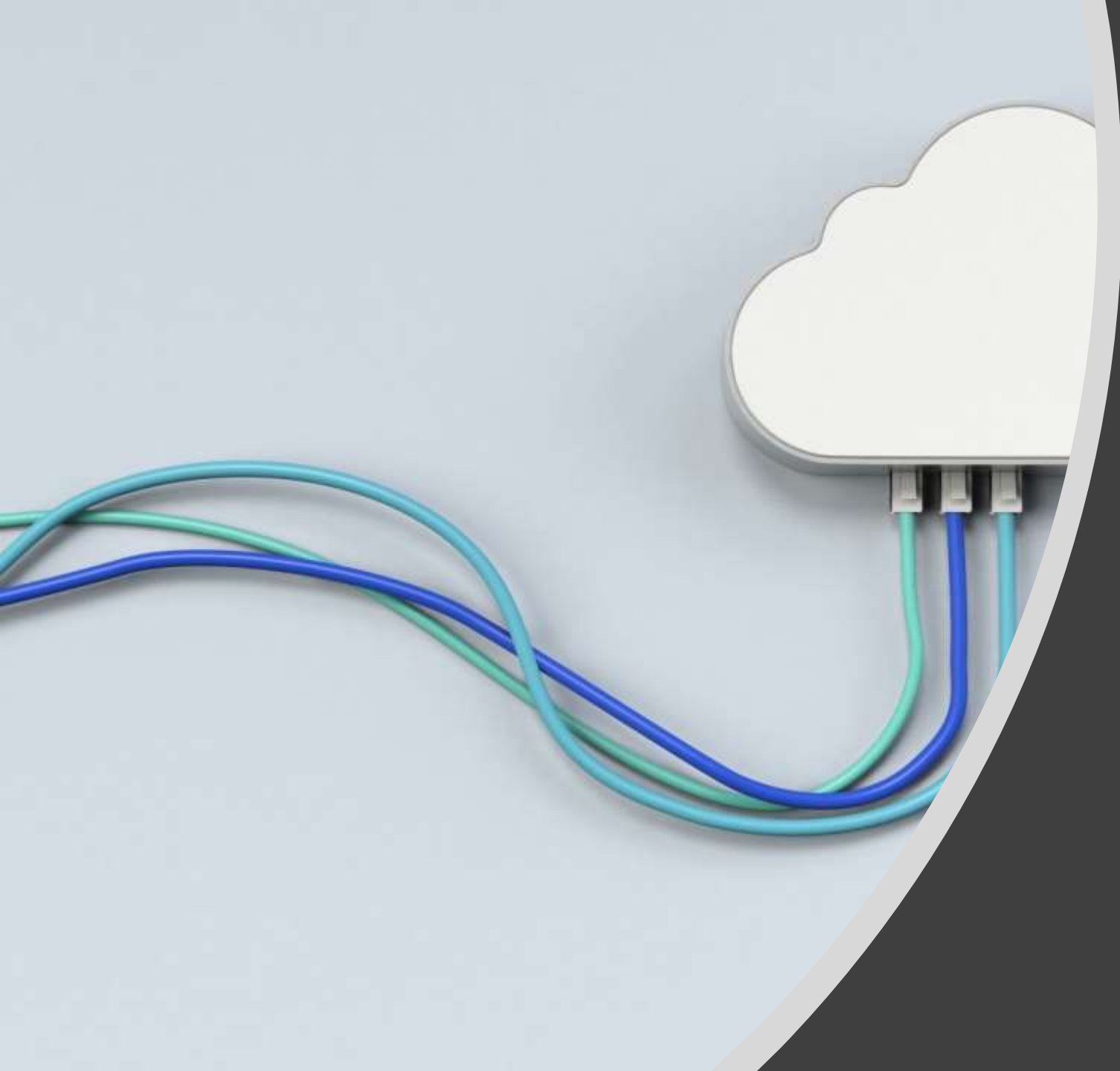
Updated 7:20 PM ET, Wed October 7, 2020



**New York (CNN Business)** – Federal banking regulators will fine Citibank \$400 million for shortcomings in its risk management and other internal controls processes.

The Office of the Comptroller of the Currency, an agency within the Treasury Department, cited "serious and longstanding deficiencies and unsafe or unsound practices" in Citibank's risk management and data governance.

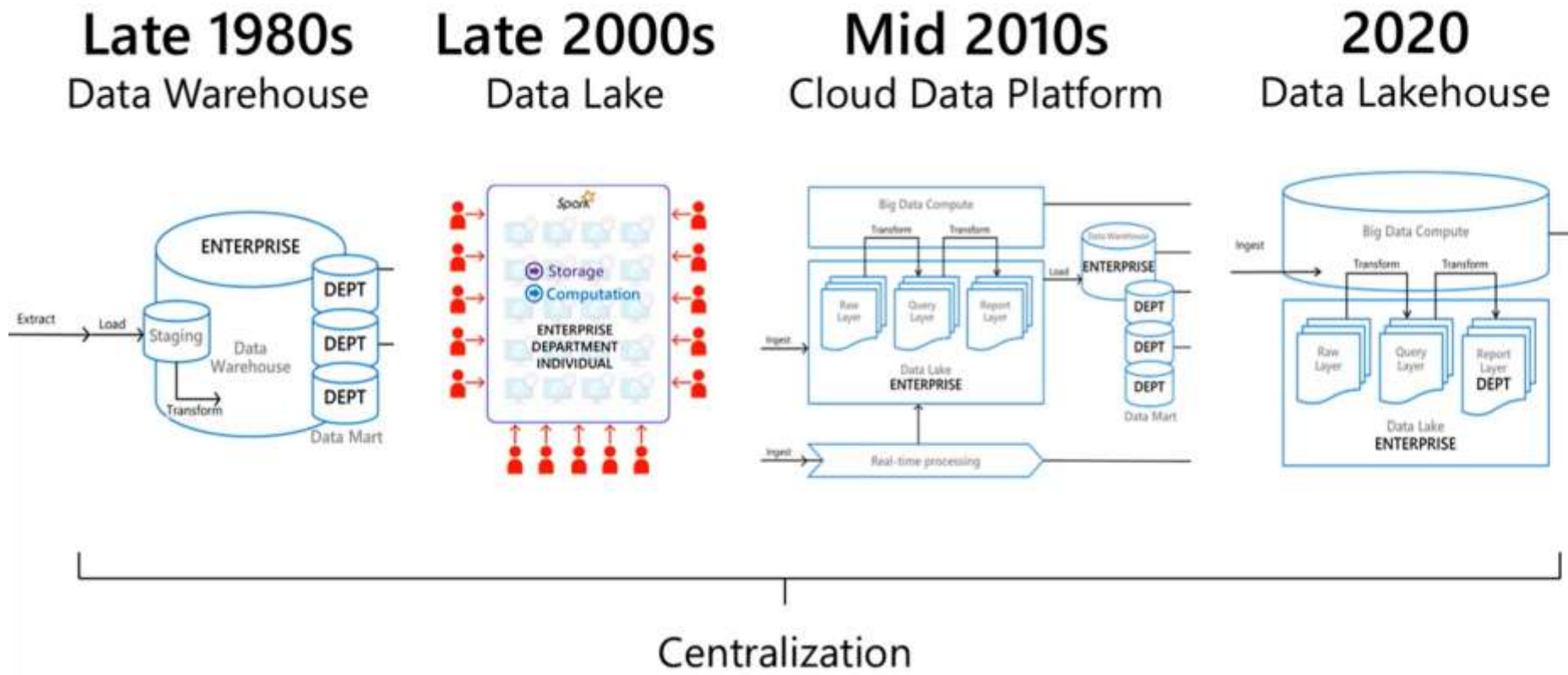
"The OCC took these actions based on the bank's longstanding failure to establish effective risk management and data governance processes and internal controls," the agency said in a release.



# What is Data Mesh Architecture?

<https://docs.microsoft.com/en-us/azure/architecture/framework>

# The Evolution of Data Architecture



**Emerging**

**Data Fabric  
Data Centric  
Data Mesh**



# Data Mesh Architecture Goals

Less than half of an organization's structured data is actively used in decision making. -MIT



We need to be able to react to real-time **streaming data** that may sit outside the organization's boundaries.

We each create PBs of data every day, most of which is unstructured and 99% of which will never be used - Forbes



We don't want to spend time structuring or replicating **huge volumes** of data, 99% of which we will never use.

By 2025, 75 percent of data will live at the edge - Gartner



We need to be able to consume and join data sets from a variety of **disparate sources** including data at the edge.

---

**To leverage big data for fueling innovation we need a flexible, scalable, decentralized platform that treats data as a consumable product.**

*So how does the Data Mesh paradigm accomplish this?*



# Research Feedback

Lack of data ownership

Lack of data quality

Difficult to see interdependencies.

Model conflicts across business concerns.

Datawarehouse serves as large integration Database.

Siloed teams => Business and IT work in silos

Disconnect between the data producer's vs data consumers

Central team becomes the bottleneck

Lack of Scale (Organizational)

Lack of prescribed guidance from Microsoft

Small changes become too risky due to unexpected consequences

Integration into IT Service Management Solution

# Data Mesh



Zhamak Dehghani

Data Mesh Architecture was first introduced in 2019 by ThoughtWorks via the blog *How to Move Beyond a Monolithic Data Lake to a Distributed Data Mesh*.

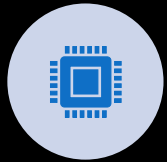
Data mesh is an architectural and organizational governance paradigm that challenges the age-old assumption that we must centralize data, have it all in one place or have data managed by a centralized team to deliver value.

Sources:

[ThoughtWorks Data Mesh Seminar at QCon](#)

[How to Move Beyond a Monolithic Data Lake to a Distributed Data Mesh](#)

# Four Principles of Data Mesh



**Domain-oriented, Decentralized Data Ownership With A Centralized Governance Hub**



**Data As A Product**



**Self-Serve Data Infrastructure As A Platform**



**Federated Computational Governance Platform**

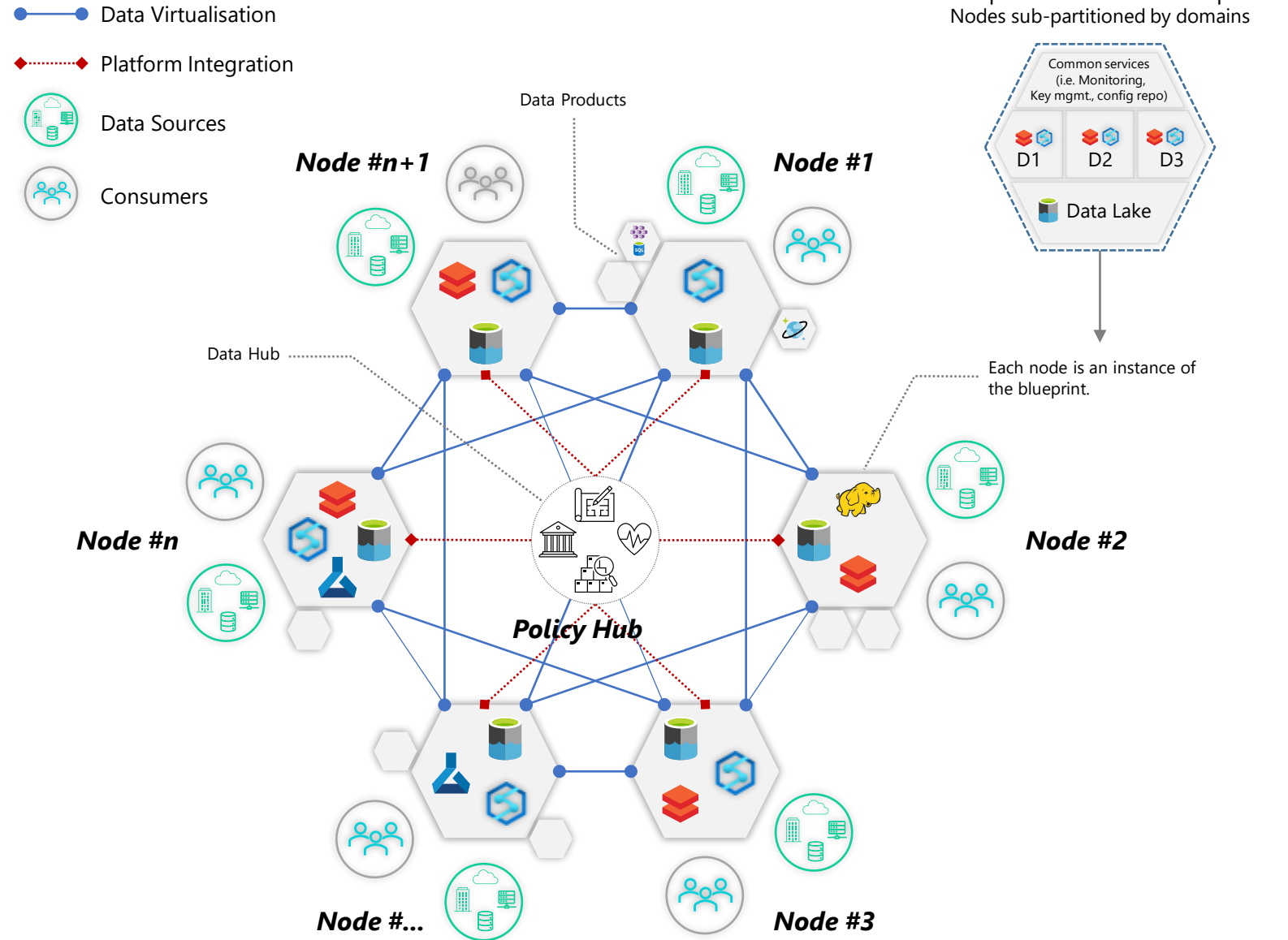
Data Mesh Architecture proposes a **decentralized architecture** where data sets are **domain-driven** and **treated as products**. These products are owned by domain teams that intimately know their data and are delivered via a **self-serve data platform**. We allocate specific roles with accountabilities to provide data as a product, abstracting away the technical complexity into the self-serve infrastructure layer so we can create these products more easily and more rapidly. To get value from data products they must be interoperable through global standardization implemented and enforced by a **centralized computational governance platform**.



# Domain-Oriented, Decentralized Data Ownership And Architecture With A Centralized Governance Hub

# Harmonised Mesh

- Azure Harmonised Mesh allows multiple groups within an organisation to operate their own analytics platforms whilst adhering to common policies and standards.
- The central datahub hosts data catalogue, mesh wide audit capabilities, monitoring and auxiliary services for automation.
- The central data platform group in the hub defines blueprints that encompass baseline security, policies, capabilities and quality standards.
- New nodes are instantiated based on these blueprints, which encompass key capabilities to enable enterprise analytics (i.e.. Storage, monitoring, key management, ELT, analytical engines, and automation)
- Node instances can be augmented to serve respective business requirements, i.e. deploying additional domains, customising domains and data products within the node.
- Nodes are typically split by either org-division, function, or region.



*Acquired companies' data can be placed in its own node*  
*Multi-cloud data can be placed in its own node*



# Azure Tools and agility

Governance Hub tools on the bottom row and Product Node tools above



Azure Data Factory



Azure Import/Export service



Azure CLI



Azure SDK



Azure IoT Hub



Azure Event Hubs



Kafka on Azure HDInsight



Azure SQL DB



Azure Cosmos DB



Azure Synapse Analytics



Azure DevOps



Azure Analysis Services



Power BI



Azure Blob Storage



Azure Data Lake Store



Azure Data Lake Analytics



Azure HDInsight



Azure Databricks



Azure ML



ML Server



Azure Databricks



Azure Search



Azure Purview



Azure Stream Analytics



Azure Data Share



Bot service



Cognitive Services



Azure Blueprint



Azure Purview



Azure Automation



Azure ExpressRoute



Azure Active Directory



Azure network security groups



Azure Key Vault



Azure API Management



Azure Security Center



Operations Management Suite



Azure Firewall



Azure Monitor



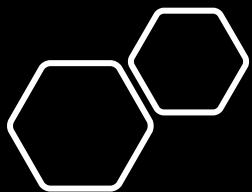
Azure Sentinel



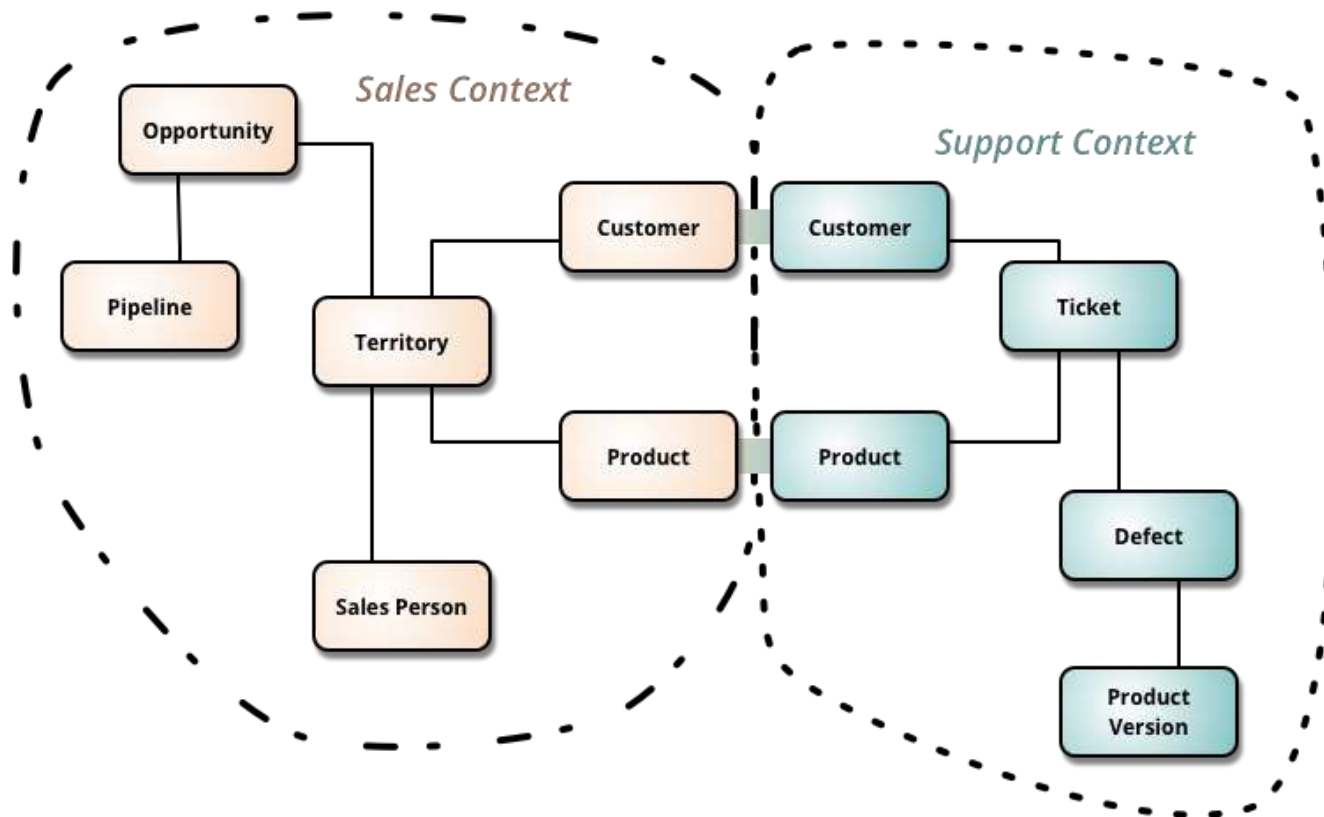
Azure Cost Management



Azure Arc



# Domain Driven Design Bounded Contexts





# Serve the Needs of Multiple Personas

## Hub Personas:

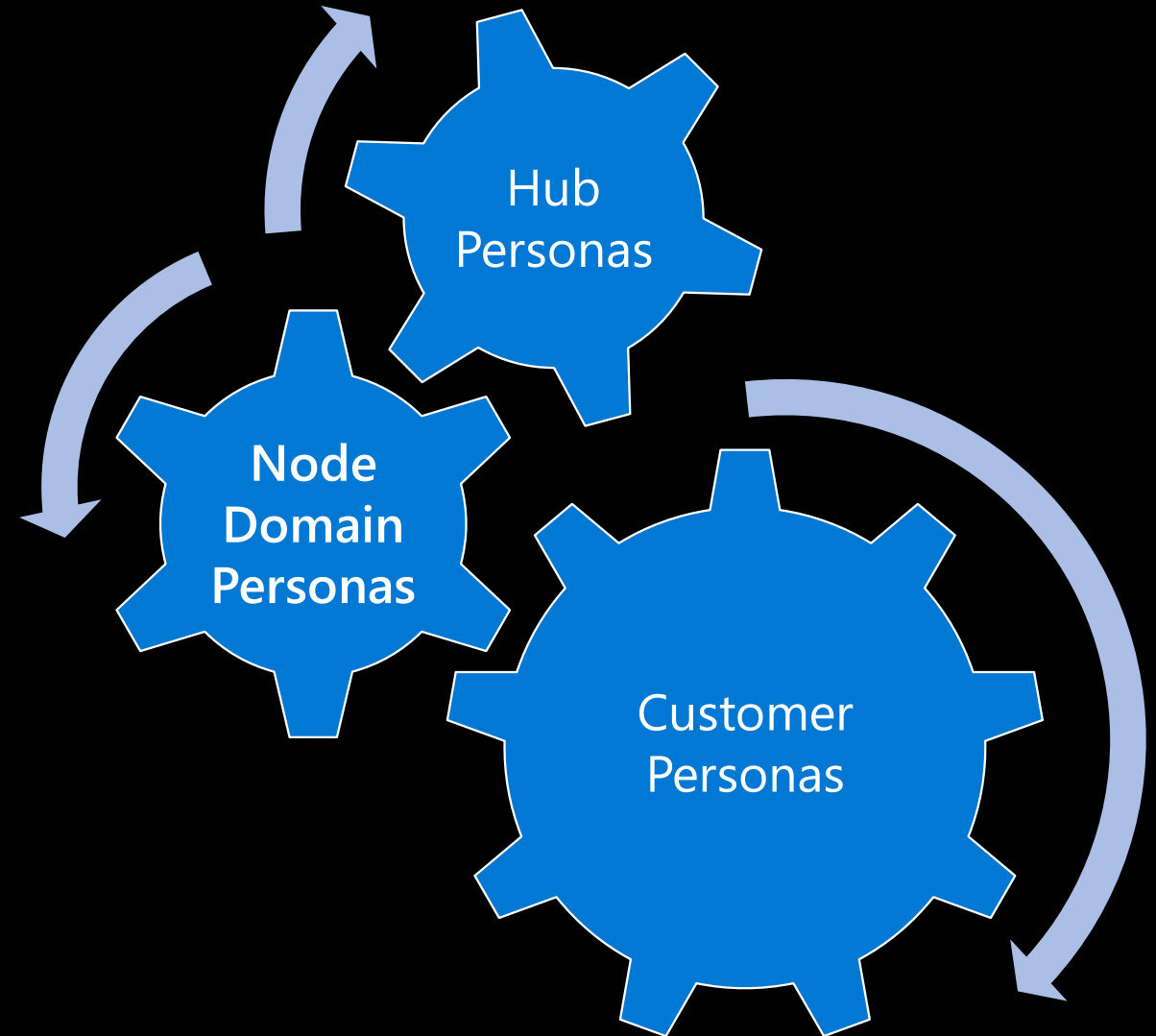
- Infra engineer
- SecOps engineer
- Networking engineer
- Cloud architect

## Node Domain Personas:

- Data engineer
- Software engineer
- DevOps engineer
- Product owner

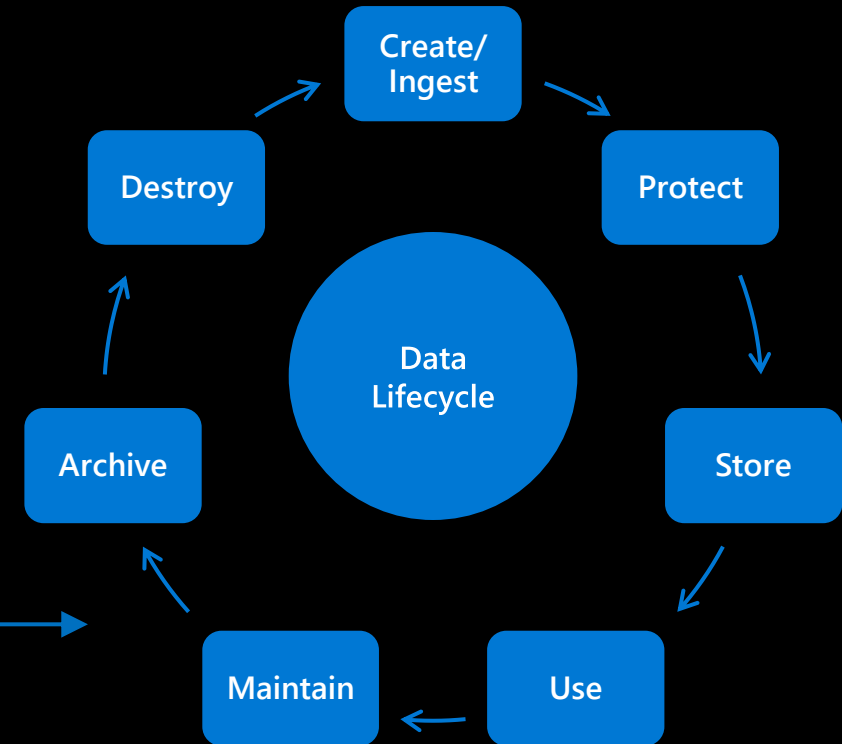
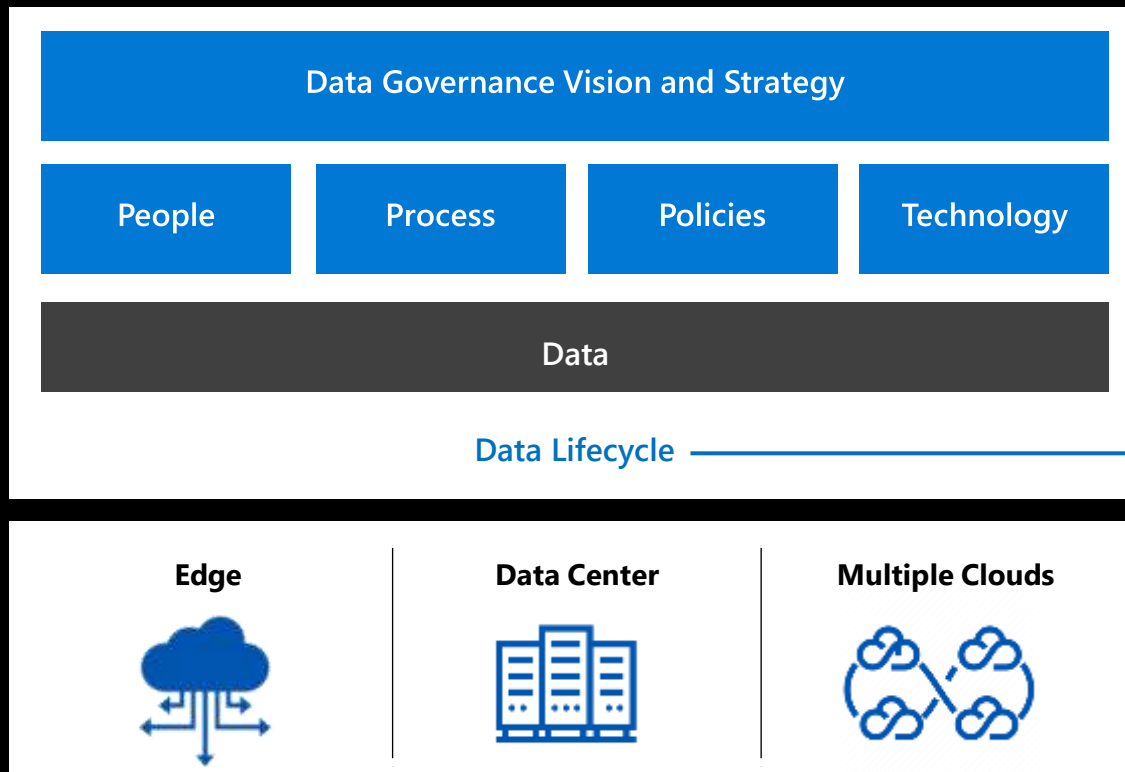
## Customer Personas:

- Citizen data scientist
- Data scientist
- Software developer
- Business analyst
- Executive report consumers
- Downstream applications



# What about Governance?

**Governance Hub:** Data product quality standards, blueprints with policies for quickly deploying new nodes, centralized security model, SLAs. RBAC and Access Control happens in the Hub.



**Domain Nodes:** Data governance, data management, API usage documentation, SLOs



# Data as a Product

Microservices • SLOs/SLAs • Consumable APIs • Cross-Functional Teams

# Domain Data as a Product

Non-negotiable usability characteristics



## **DISCOVERABLE & UNDERSTANDABLE**

Via data catalog or glossary we must be able to understand source, owners and lineage.



## **SELF-DESCRIBING SEMANTICS**

Well-described semantics and syntax ideally accompanied by sample datasets. Data schemas are a starting point.



## **ADDRESSABLE**

Data must be easily accessible once discovered. Different domains might store and serve their datasets in different formats, but open formats are required (Parquet).



## **INTEROPERABLE**

We must be able to correlate, join and synthesize data across multiple domains. This is the foundation for building distributed systems.



## **TRUSTWORTHY & TRUTHFUL**

Inconsistencies or missing data must be documented. Data owners should provide service level objectives around truthfulness of data and how closely it mirrors events.



## **SECURE (GOVERNED BY RBAC)**

Accessing product datasets securely is a must. Access control is applied at a fine granularity for each domain product. Access control policies can be defined centrally but applied at the time of access to each individual dataset product.

# Developed by Cross-Functional Teams

Domains that provide data as products need to be augmented with new skill sets and roles.

## Data Product Owner:



Makes decisions around the vision and the roadmap for the data products as well as the lifecycle of data assets.  
Owns the satisfaction of data consumers and continuously measures and improves the quality of the data the domain owns and produces.  
Defines success criteria SLOs and business-aligned Key Performance Indicators (KPIs) for their data products.

## Data Engineer/Data Product Owner:



Responsible for developing, serving and maintaining the domain's data products.  
Continuous delivery and automate testing, when it comes to building data assets.  
Documentation and stewardship over data.

## Infra/Sec Engineers:



Manage data infrastructure and security.  
Integrate the domain with the policies and governance prescribed by the hub.  
Implement HR and DR strategy, run threat hunting drills.

## Software Engineer:



DevOps practices infused into data pipelines and APIs using microservices architectures.  
Documentation and stewardship over APIs.  
Data must be treated a foundational piece of any software ecosystem, hence software engineers.

## Product Development Mindset and Processes

Domain Driven Design  
Agile  
DevOps  
Microservices

# Characteristics of Microservices

**Small and focussed:** The “micro” in microservices is about scope, not size and is focussed towards a specific problem.

**Bounded Context:** Microservices are independent of the underlying architecture of other microservices.



**Language neutral:** Different microservices application can be written in a different programming language and communication with microservices is done through REST API (an HTTP-based resource API).

**Loosely coupled:** This essential characteristics enables organizations to deploy rapidly and frequently

**Multiple code base:** Each microservice has its own code base

# BENEFITS OF A MICROSERVICES ARCHITECTURE



**CONTINUOUS DELIVERY  
AND DEPLOYMENT**



**BETTER  
SCALABILITY**



**IMPROVED FAULT  
ISOLATION**



**GREATER  
FLEXIBILITY**



**SMALLER  
DEVELOPMENT TEAMS**



**HIGHER SOFTWARE  
TESTABILITY**



**IMPROVED  
MAINTAINABILITY**



# Team Accountabilities

## Success criteria for Hub



How quickly a new domain can be up and running with RBAC and everything ready to go



How seamlessly the hub can unify data access control (RBAC, policy)



Enforce SLAs

## Success criteria for Nodes



Decrease lead time for someone to find the data, make sense of it and use it



Delight of customer demonstrated in growth in number of users



Uptimes and SLOs



# Self-Service Data Platform

Serves Diverse Personas • Data Catalog • Data Marketplace



# Serving Needs of Data Consumers

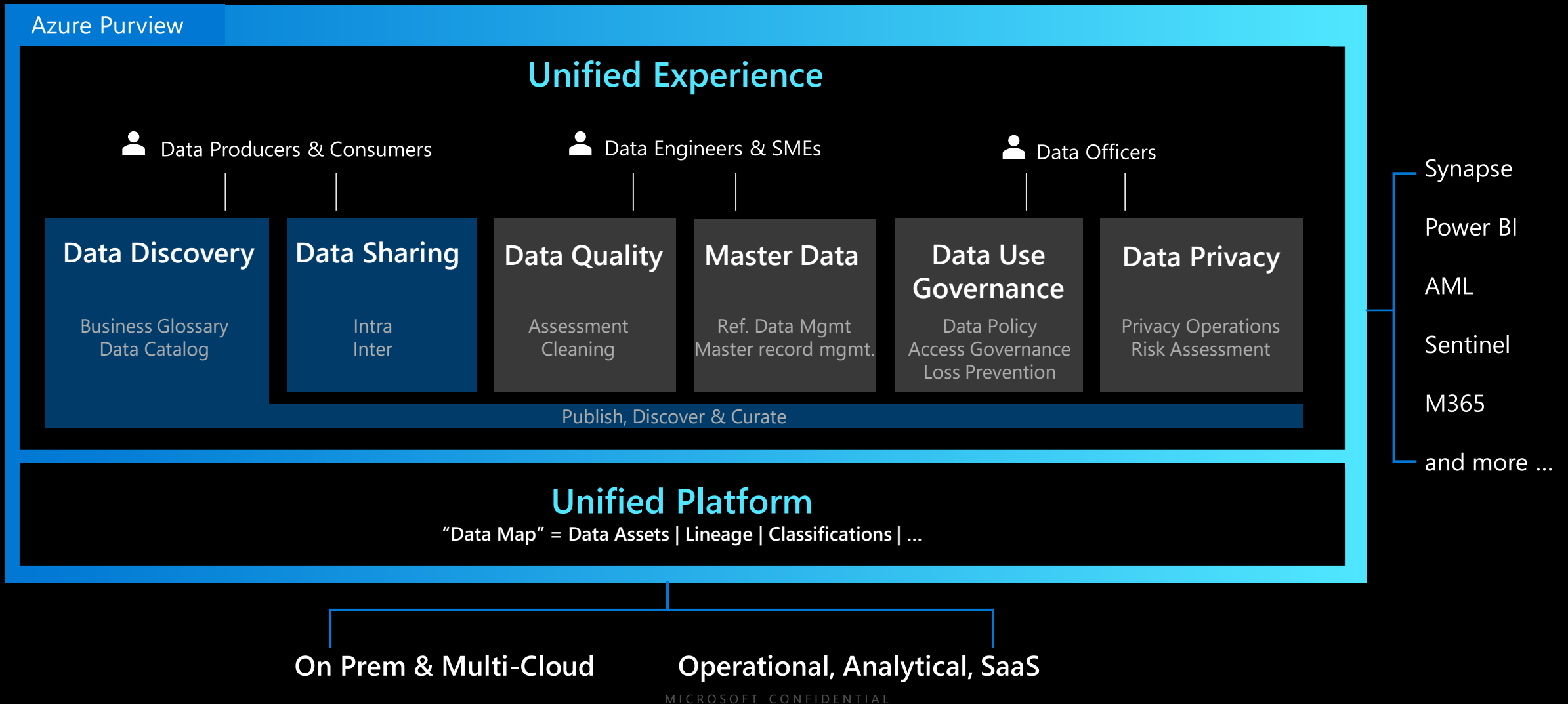
**There is a long list of capabilities that a self-serve data infrastructure as a platform provides to its users, including:**

- Scalable polyglot big data storage
- Datasets are discoverable
- Encryption for data at rest and in motion
- Data product versioning
- Data product schema
- Data product de-identification
- Unified data access control and logging
- Data pipeline implementation and orchestration
- Data product discovery, catalog registration and publishing
- Data governance and standardization
- Data product lineage
- Data product monitoring/alerting/log
- Data product quality metrics (collection and sharing)
- In memory data caching
- Federated identity management
- Compute and data locality

# Azure Purview for Data Governance

A unified approach to data governance & stewardship

■ Roadmap



# Federated Identity Management: Active Directory

Reduce sign-in friction



Single sign-on

Enhance security



Identity Protection

Leverage M365



Microsoft Graph

Comply with IT



Publisher verification  
User provisioning

Integration scenarios

# PowerBI – Data Mesh BI Solution for All Personas

## Software Engineer / Data Scientist:

- PowerBI Embedded Apps
- PowerBI R and Python Custom Visuals
- PowerBI D3.js Custom Visuals

## Citizen Data Scientist:

- DataFlows
- Cognitive Services
- Azure Machine Learning
- AI Powered Visualizations

## Data Engineer:

- Tabular, composite and semantic modeling with XMLA Endpoint and Azure Analysis Services feature integration

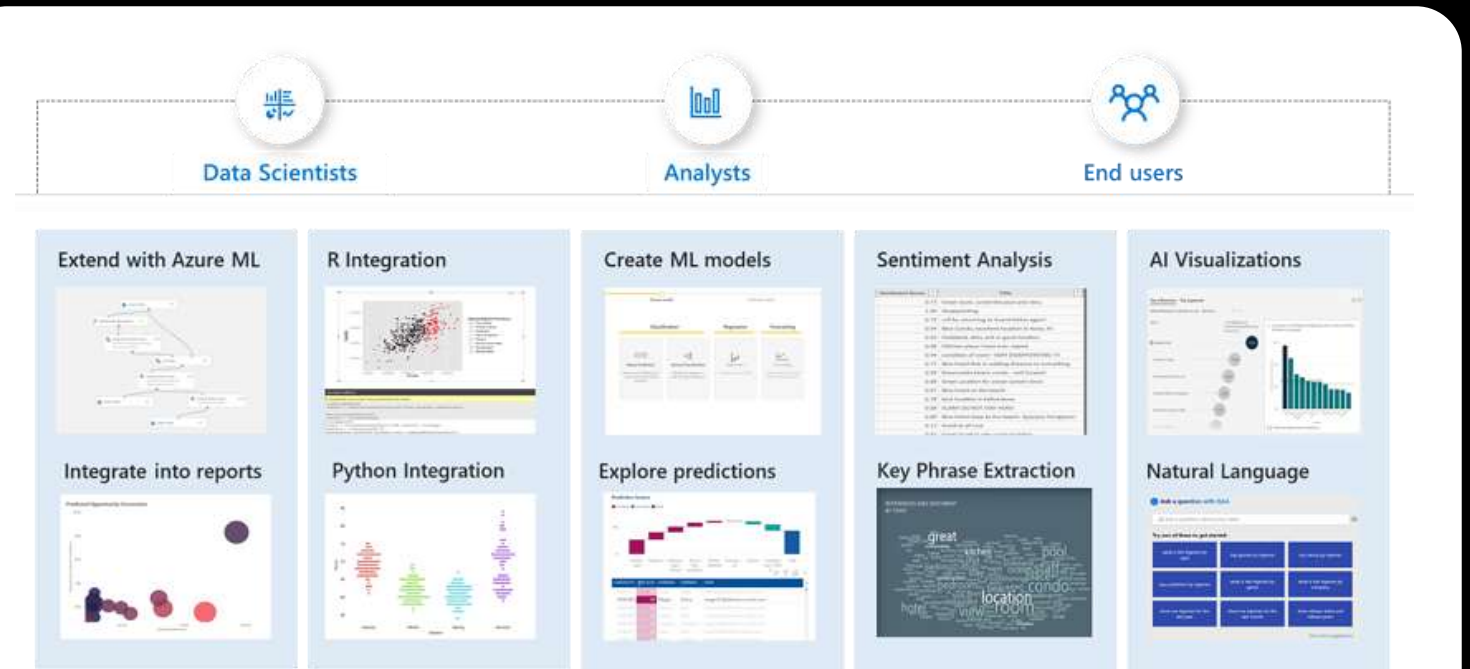
## Business Analyst:

- PowerBI Reports
- PowerBI Apps

## Data Steward:

- Fully integrated with Azure Purview
- Data Lineage
- Data classifications and sensitivity labels

Not all customers are data scientists,  
data engineers or software developers



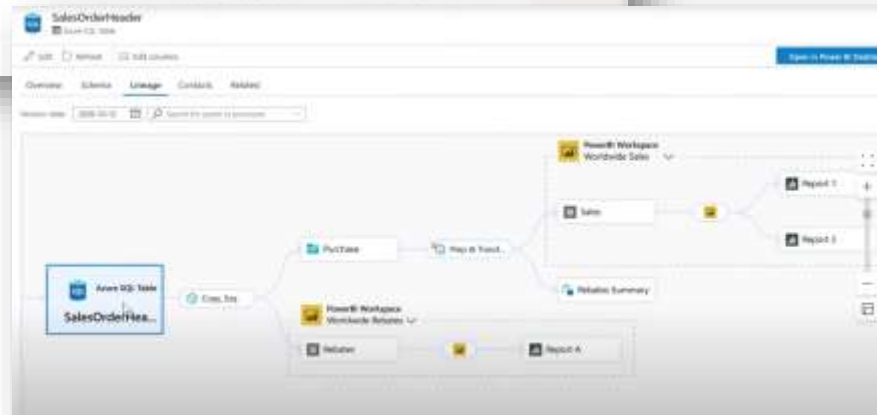
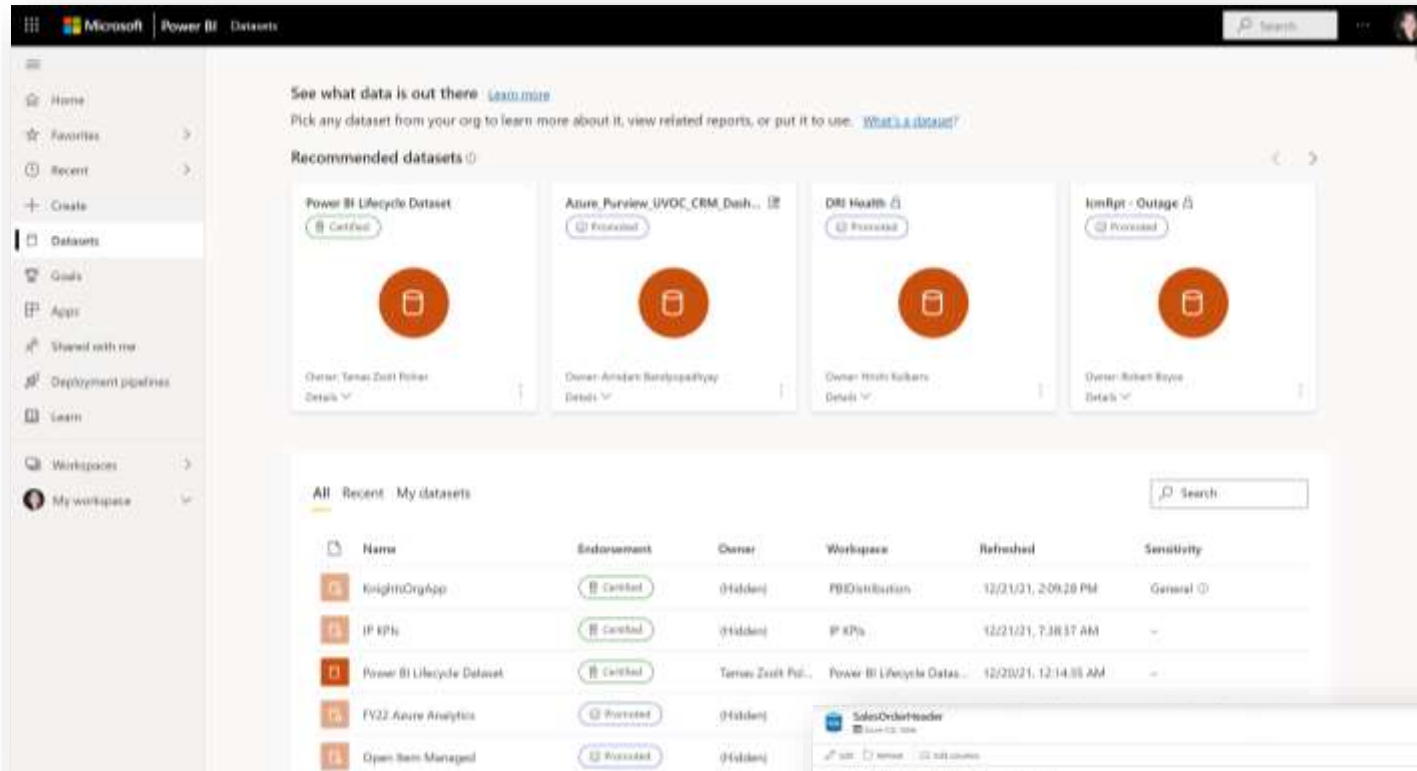
**PowerBI Apps • Datasets • Reports • Dashboards • Visualizations**

<https://www.bluegranite.com/blog/artificial-intelligence-features-in-power-bi>

[PowerBI Features As of Apr 2021](#)



# PowerBI Datastore From PBI Service

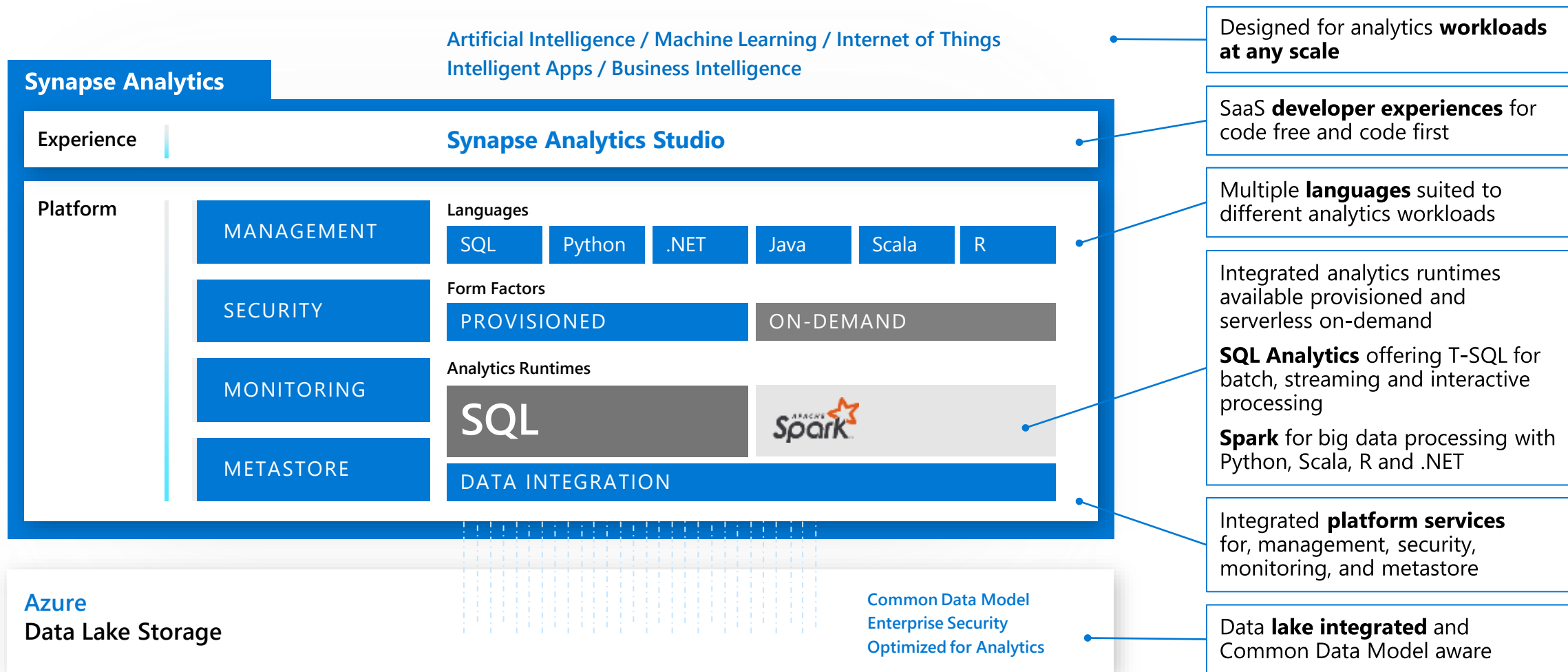


- Create a data marketplace with Certified and Recommended Reports, Dashboards and Datasets
- Share reports and dashboards B2B and B2C
- Leverage Lineage Functionality within PowerBI or Purview to View Lifecycle of Datasets, reports and dashboards



# Azure Synapse Analytics for Data Engineers

Integrated data platform for BI, AI and continuous intelligence

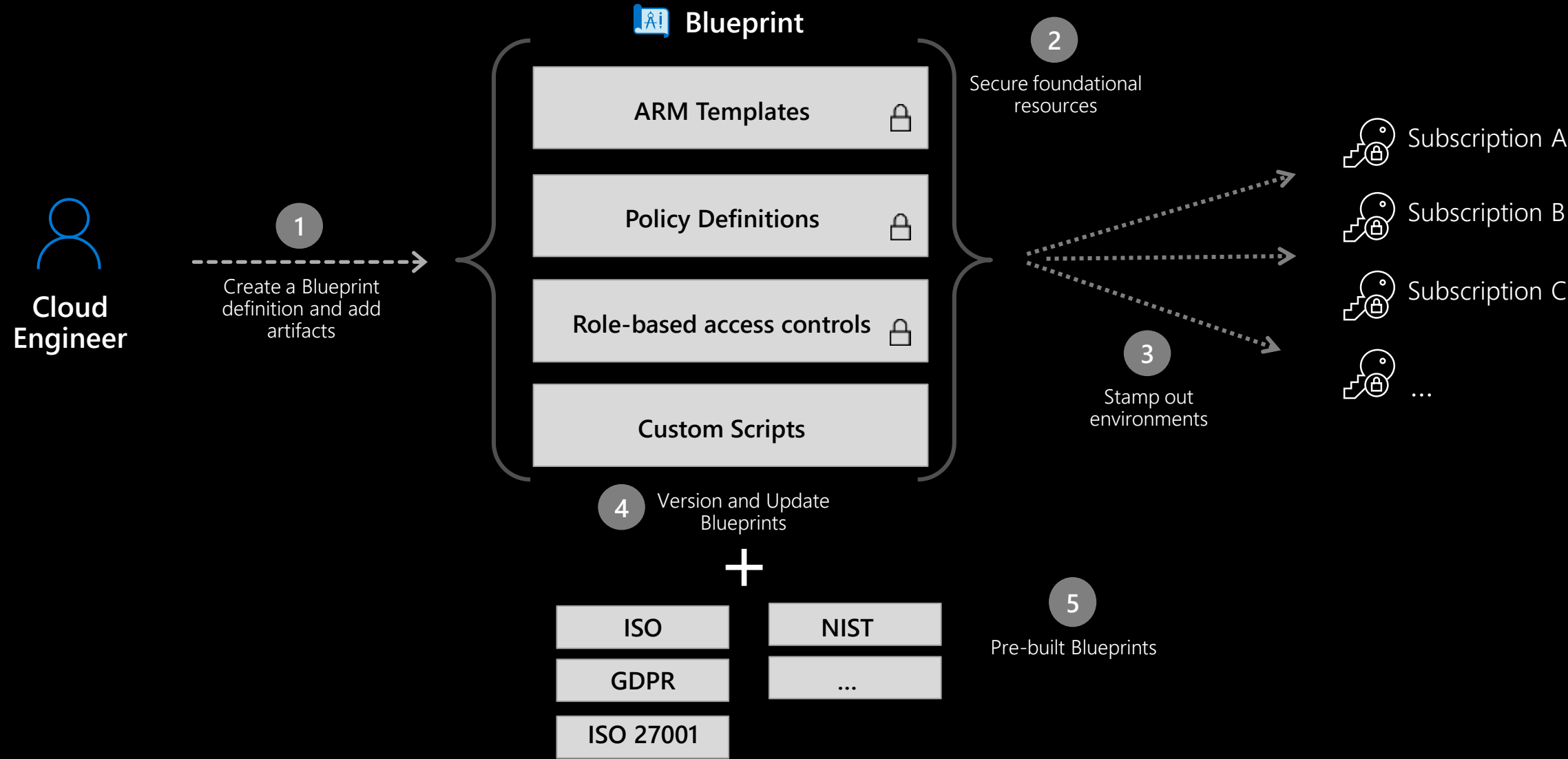




# Federated Computational Governance

Secure • Open-Source • Portable • Interoperable • Multi-cloud • Hybrid

# Azure Blueprints



# Azure: Delivering Open Source Innovation in the Cloud

## Management



## DevOps & PaaS



## Applications



## Frameworks & Tools



## Data Platform



## Infrastructure



# Data Platform Interoperability – Open Source + Azure

## Ingest

Kafka  
(Event Hub)

## Store

Parquet  
(Azure Data Lake)

HDFS  
(Azure Data Lake)

PostgreSQL/  
MariaDB/  
MySQL  
(Azure Databases)

## Prep

Spark  
(Azure Synapse, Databricks)

Storm  
(Azure HDI)

ONNX  
Jupyter  
MLFlow

## Serve

Apache Hive  
(Azure HDI)

Apache Atlas  
(Azure Babylon)

Gremlin  
(Azure Cosmos)

Azure SQL  
(macOS, Linux, Windows VM)

Cloud of Your Choice  
(Azure)

# Azure open source ecosystem

Building open solutions jointly with partners, customers, and community

Joint engineering | Joint operations | Joint support



Integrated identity | Integrated security | Integrated billing

# Popular Portable Frameworks

Build advanced deep learning solutions

Use your favorite deep learning frameworks



TensorFlow



PyTorch



Scikit-Learn



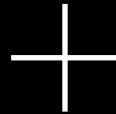
MXNet



Chainer



Keras



Without getting locked into one framework



Community project created by Facebook and Microsoft

Use the best tool for the job. Train in one framework and transfer to another for inference



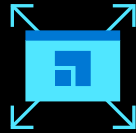


# Azure Arc

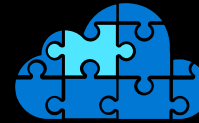
Bring Azure services and management to any infrastructure



Run Azure Data  
Services anywhere



Extend Azure management  
across your environments



Adopt cloud  
practices on-premises

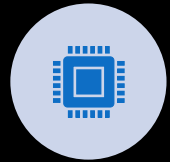


Implement Azure  
security anywhere

---

Azure Arc is a set of technologies that extends Azure management and enables  
Azure services to run across on-premises, multi-cloud, and edge

# Four Principles of Data Mesh



**Domain-oriented, Decentralized Data Ownership With A Centralized Governance Hub**



**Data As A Product**



**Self-Serve Data Infrastructure As A Platform**



**Federated Computational Governance Platform**

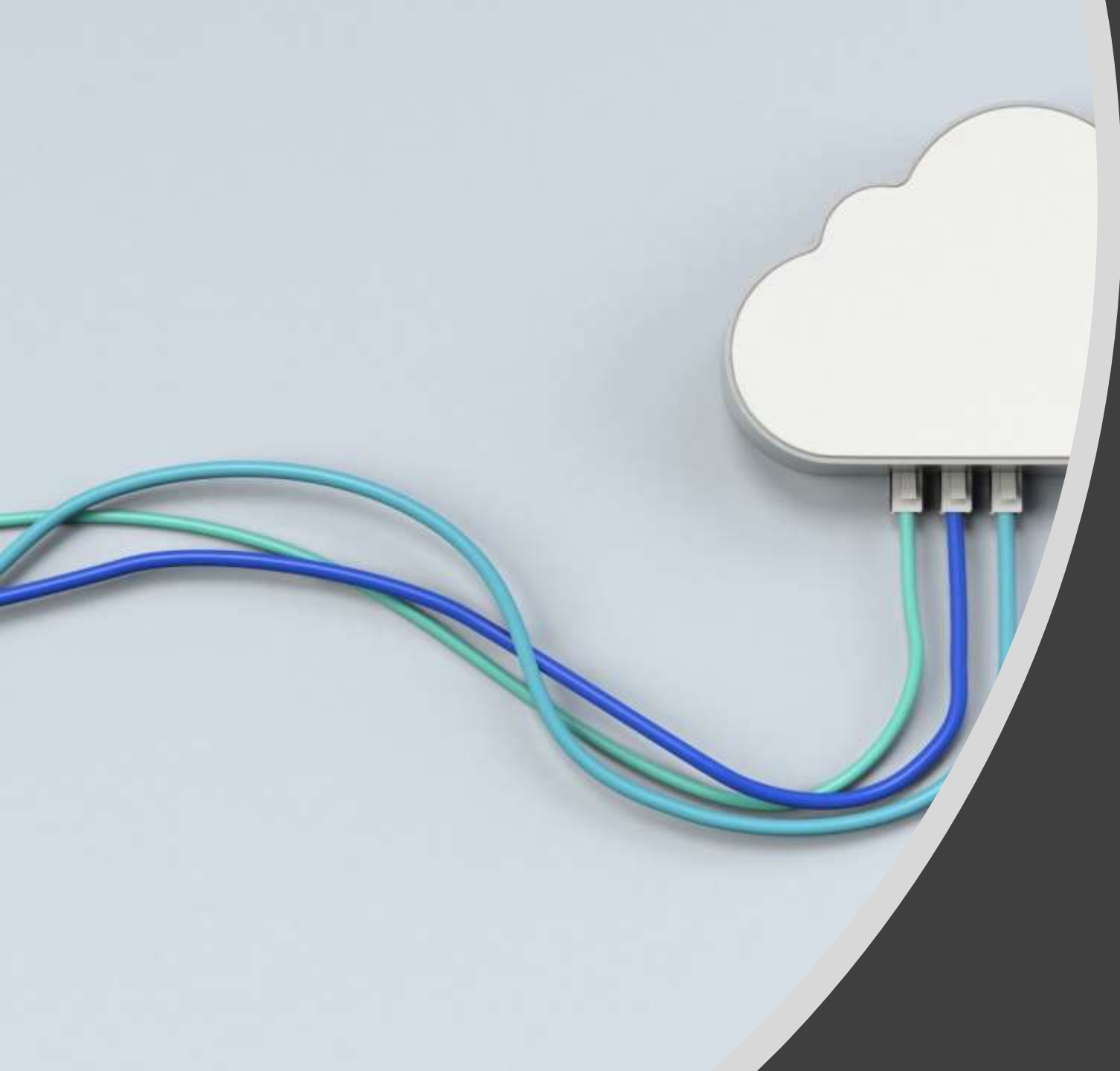
# Evaluating Hub and Node Technologies

# Key Decision Points for Governance Hub Services

1. Does my hub have easy hybrid connectivity to other clouds, on-prem data sources and third-party data centers?
2. Does my hub offer infrastructure as code with a built-in set of blueprints, policies, security and governance controls that comply with my industry and company regulations?
3. Can I govern multi-cloud data sources and on-prem data sources from my hub including cost visibility?
4. Does my hub offer an identity management solution that can easily integrate with my on-prem identity management provider and server both internal and external customers (guest users, conditional access policies, just-in-time access, etc.)?
5. Is my hub secure and does it have a SIEM?
6. Does my HUB have an industry-leading API management solution?

# Key Decision Points for Data Domain Node Services

1. Does my data service provide an open-source data format (i.e., parquet) or is it a proprietary format that will be difficult (or costly) to operate on another platform?
2. Can I access the data in a variety of different programming language and GUIs?
3. Can I serve my data and machine learning models via REST API or ONNX format?
4. Does my data store support structured, unstructured and semi-structured data?
5. Do the SLAs for my platform service provider support the SLOs I offer my customers?
6. Am I using a CI/CD tool that supports agile microservices deployments in parallel?
7. Do I have a solid data governance and data glossary tool that integrates with my data sources?
8. Are there low-code no-code options for my data consumer personas to use my data products (CEO, Business Analyst, Data Scientist)



# Data Scenarios

<https://docs.microsoft.com/en-us/azure/cloud-adoption-framework/scenarios/data-management>



# What is CAF for Data Management & Analytics



A **scalable** analytics framework designed to enable customers building an enterprise data platform.



Supports **multiple** topologies ranging across Data Centric, Data Lakehouse, Data Fabric and Data Mesh.

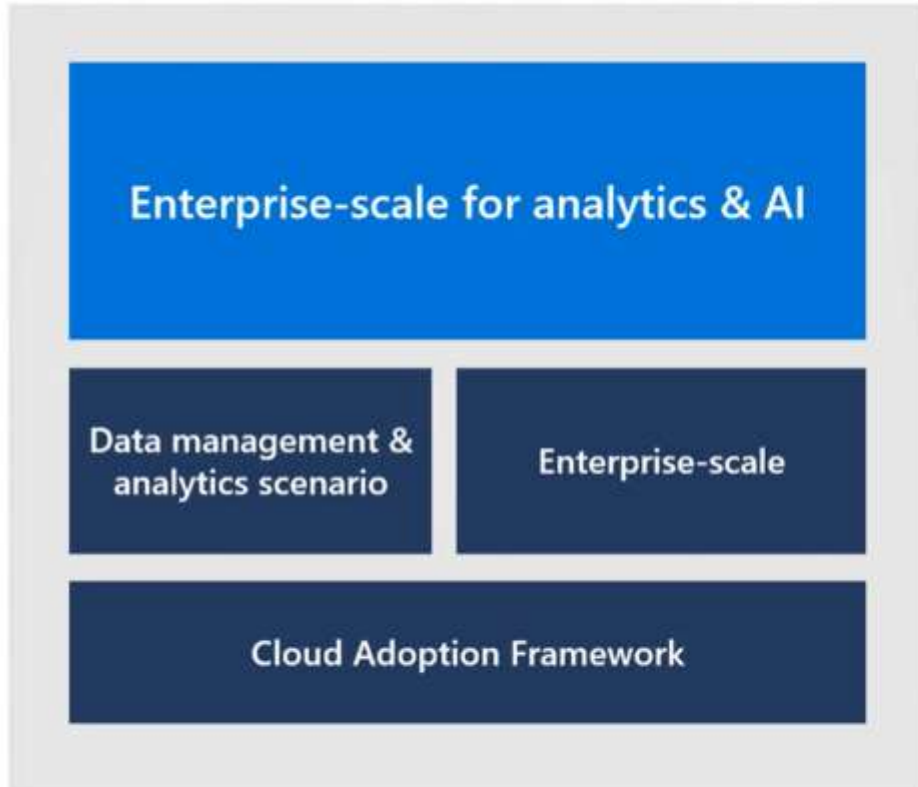


Based on inputs from a diverse international group of specialists working with a range of customers.



Published under **Cloud Adoption Framework** containing separate guidance tailored to small, medium and large enterprises.

# What is CAF for Data Management & Analytics



- ✓ Prescriptive architecture
- ✓ Designed by subject matter experts
- ✓ Documented end-to-end technical solution
- ✓ Deployment templates
- ✓ Operational usage model
- ✓ Gravitas of placing Azure at centre of data governance
- ✓ Reduce time to use cases

# CAF Data Management & Analytics Principles



Classify **governance** activities into a single landing zone called the **Data Management Landing Zone**.



Group **analytics workloads**—such as **data integrations** and **data products**—into **Data Landing Zone(s)**.



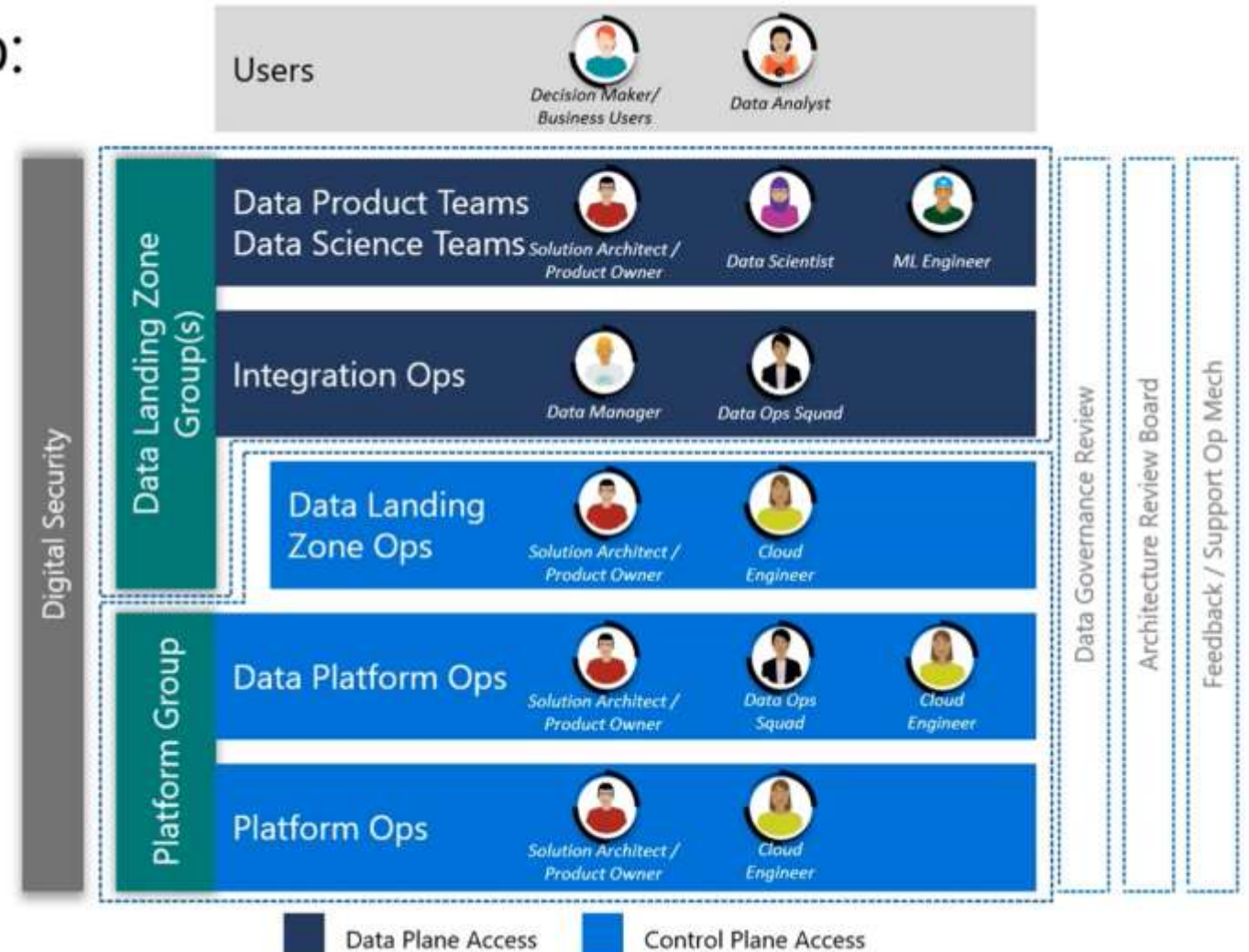
**Scale** using subscriptions by creating multiple Data Landing Zones.



Apply organizational guidance to maintain **security** boundaries.

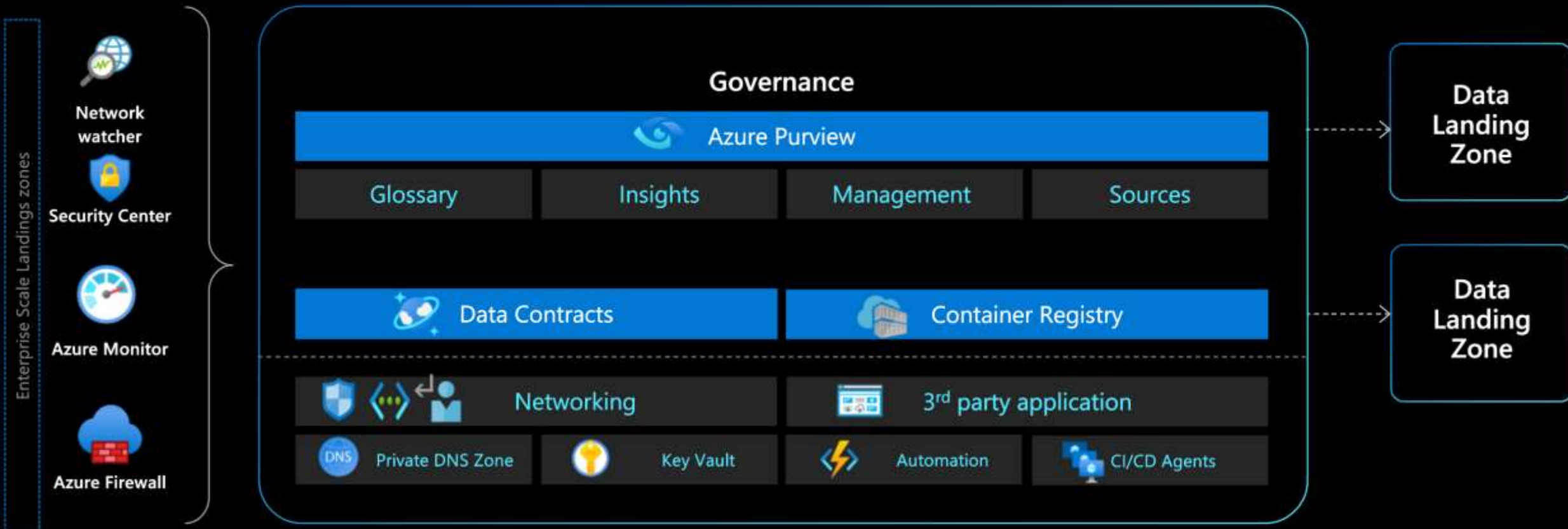
# Suggested Team Operating Model – Guidance Only

- The Data Landing Zone Group:
  - Data Product Teams (per product)
  - Integration Ops (per integration)
  - Data Landing Ops (per node)
- The Platform Group:-
  - Data Platform Ops
  - Platform Ops



# Data Management Landing Zone

Centralised governance, data cataloging, 3<sup>rd</sup> party services

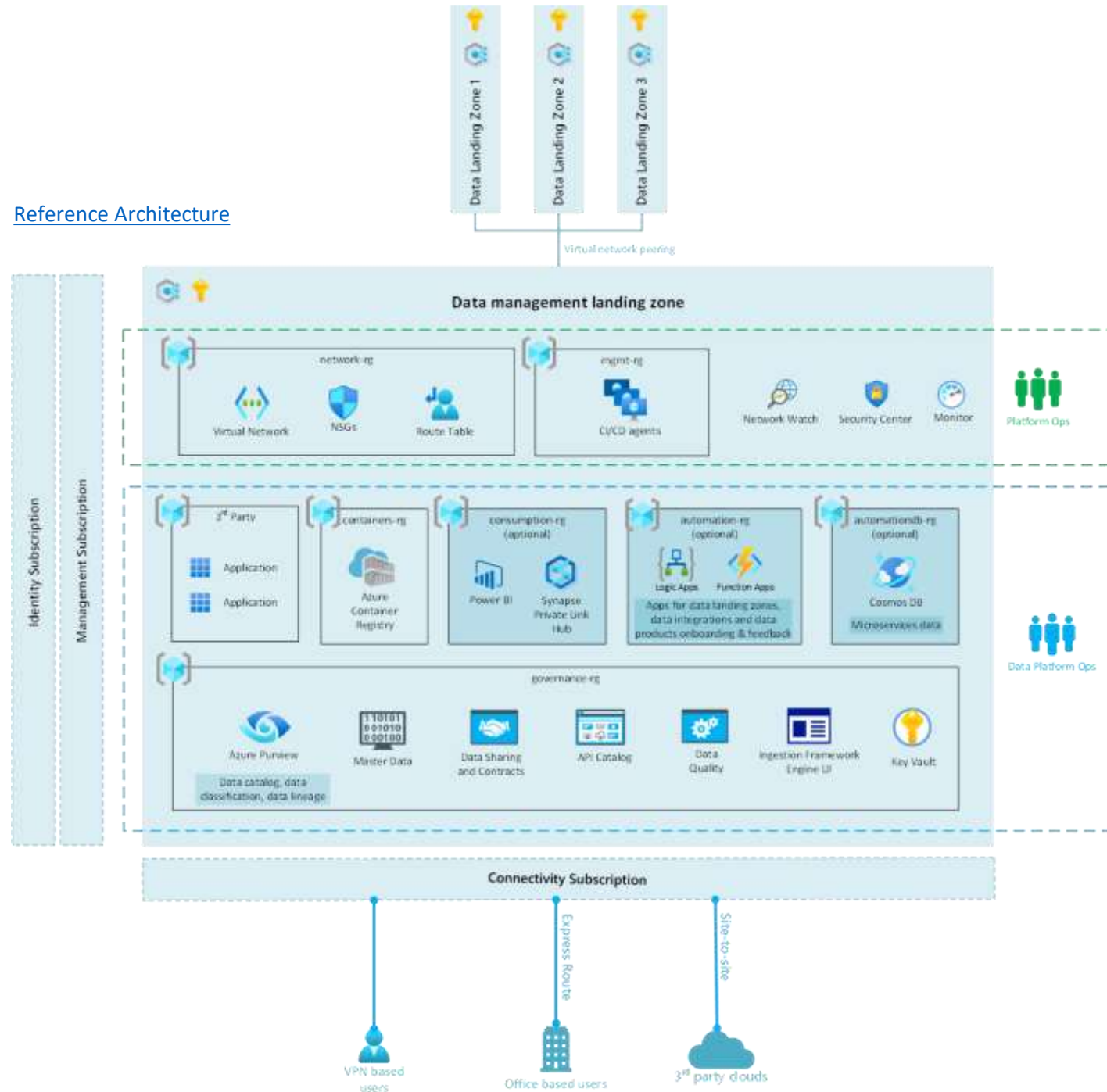




# The Data Management Landing Zone is a Centralized Governance Hub

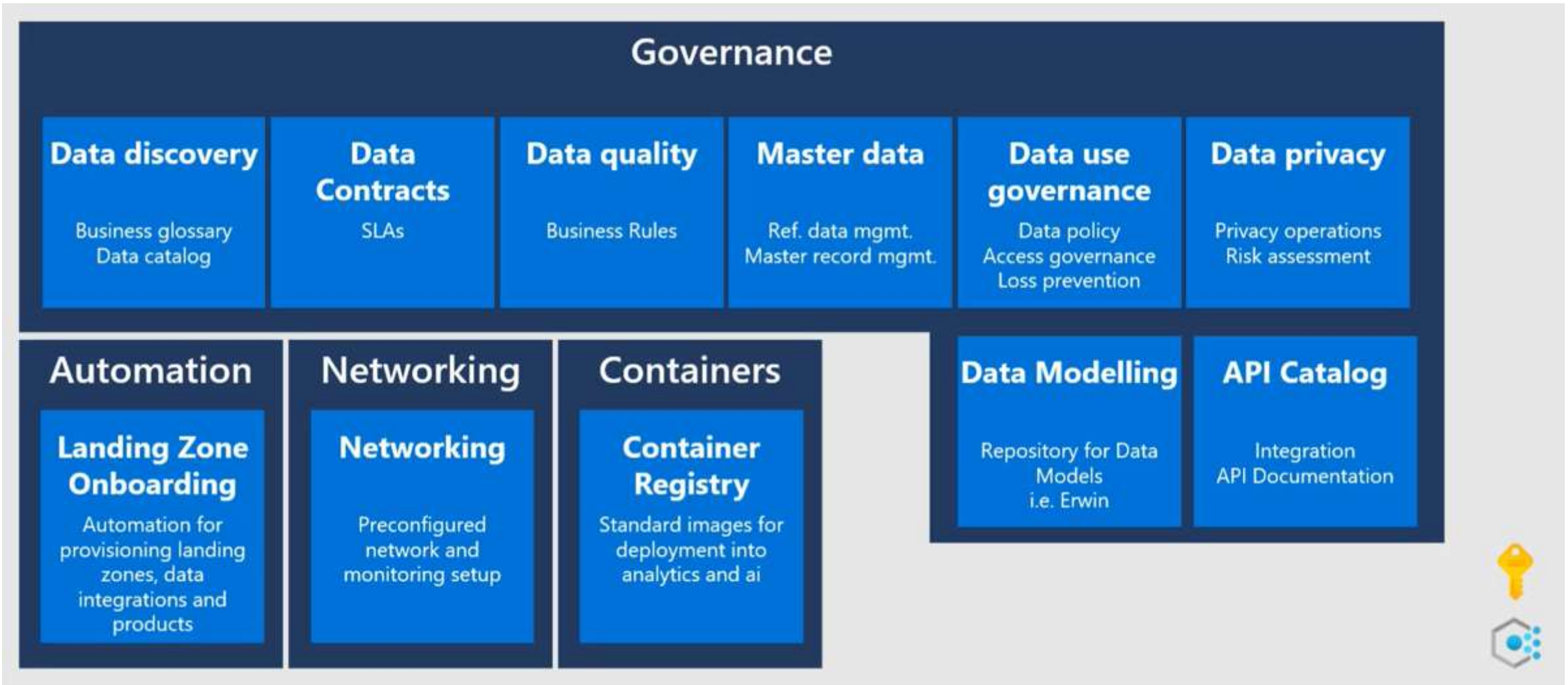
1. [Data catalog](#)
2. [Data classification](#)
3. [Data lineage](#)
4. [Data quality management](#)
5. [Data modeling repository](#)
6. [Master data management](#)
7. [API catalog](#)
8. [Data lifecycle](#)
9. [Automation interfaces \(optional\)](#)
10. [Container registry](#)
11. [Global DNS](#)
12. [Azure Synapse Private Link hub](#)

Reference Architecture



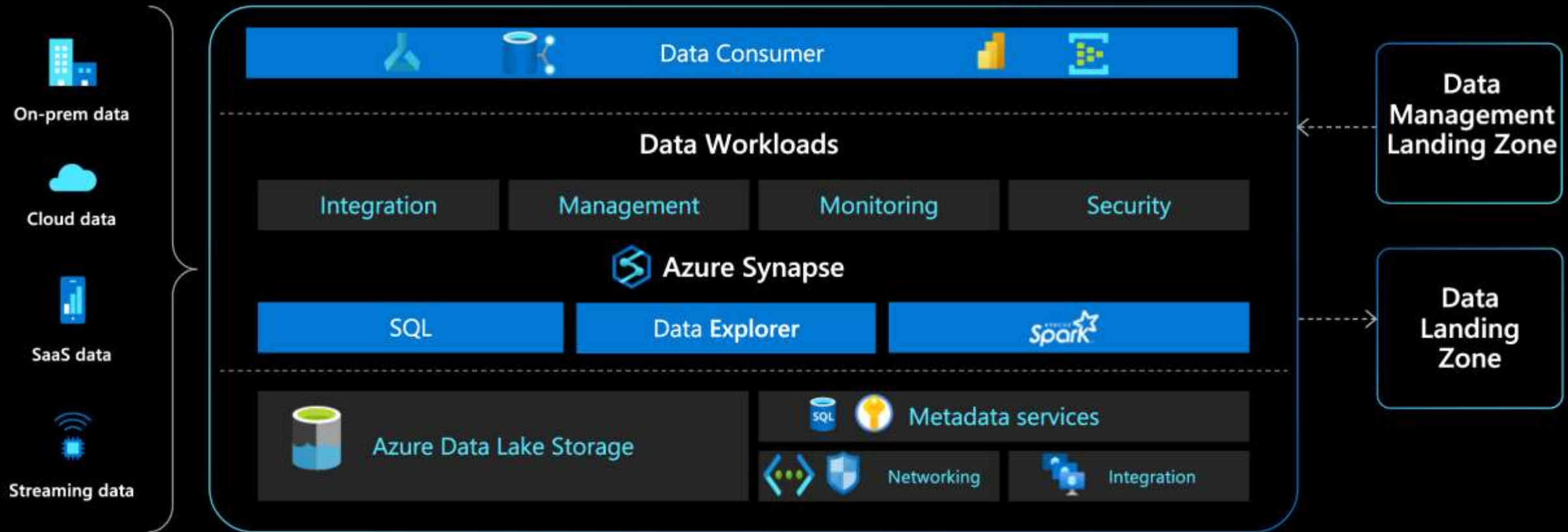


# Data Management Landing Zone

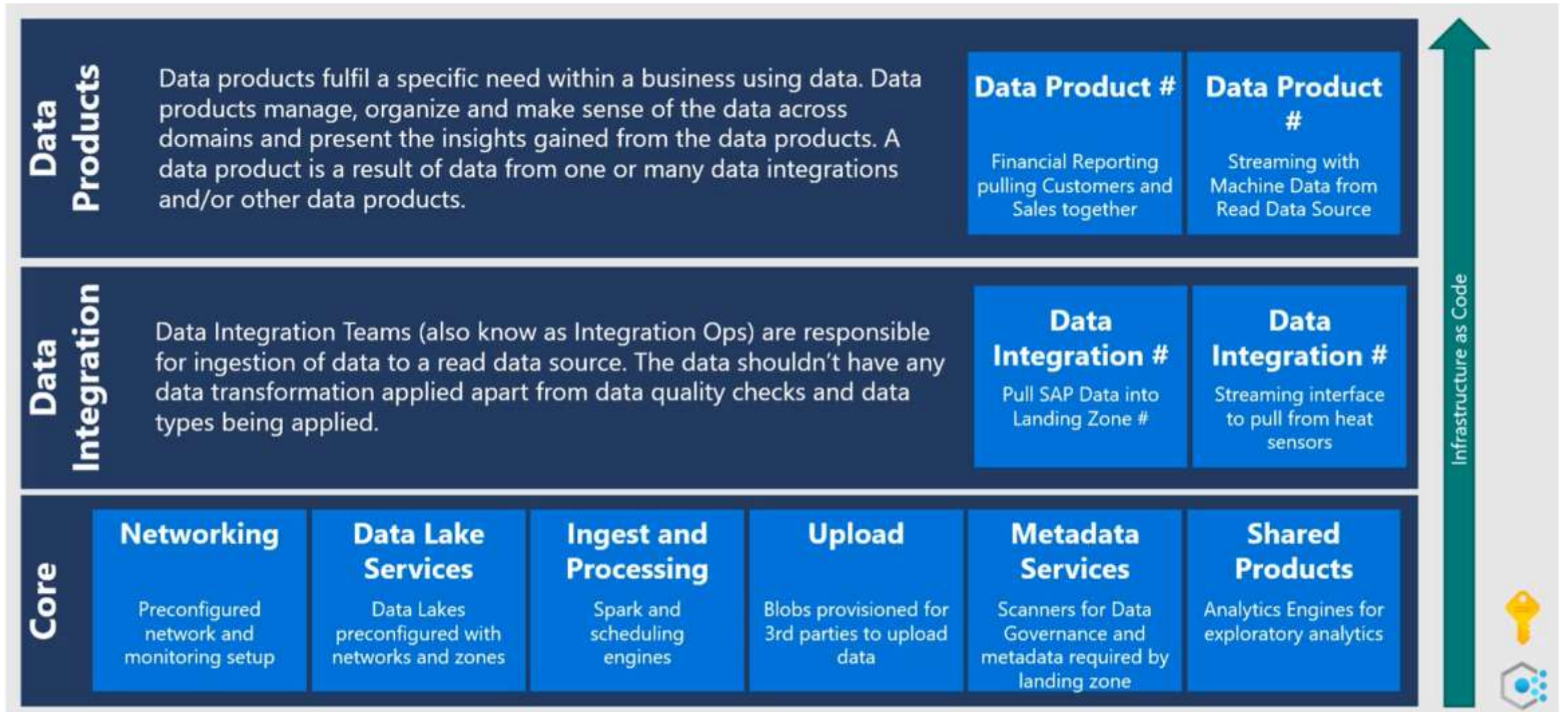


# Data Landing Zone

Limitless analytics service with unmatched time to insight



# Data Landing Zone



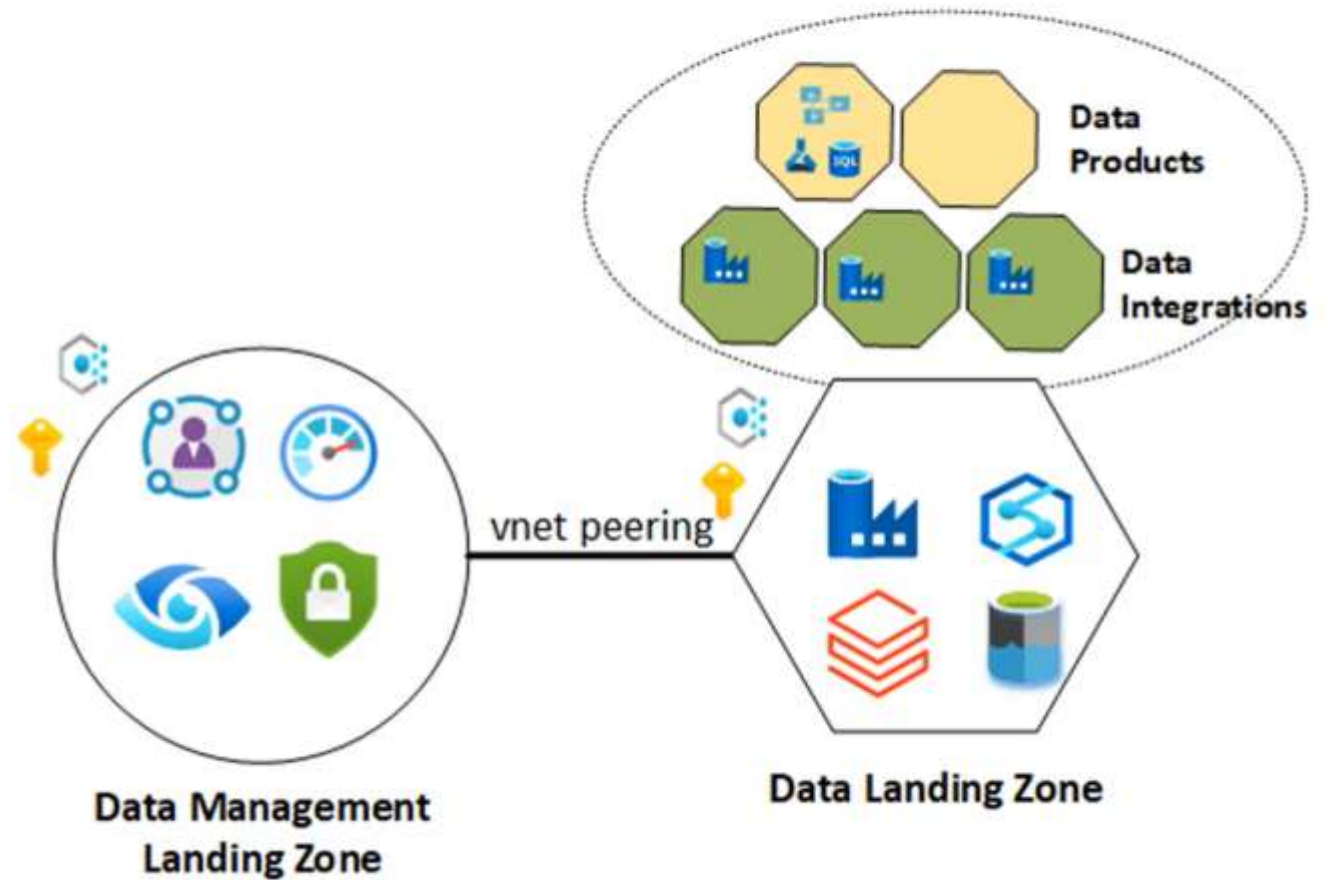
# Security and Policies

- Guidance on securing data
- Recommendations on holding PII data and highly confidential
- Azure Policies specifically for data as baseline guides for Ops Team
- Networking integration with Enterprise Scale.
- Advanced Threat Protection



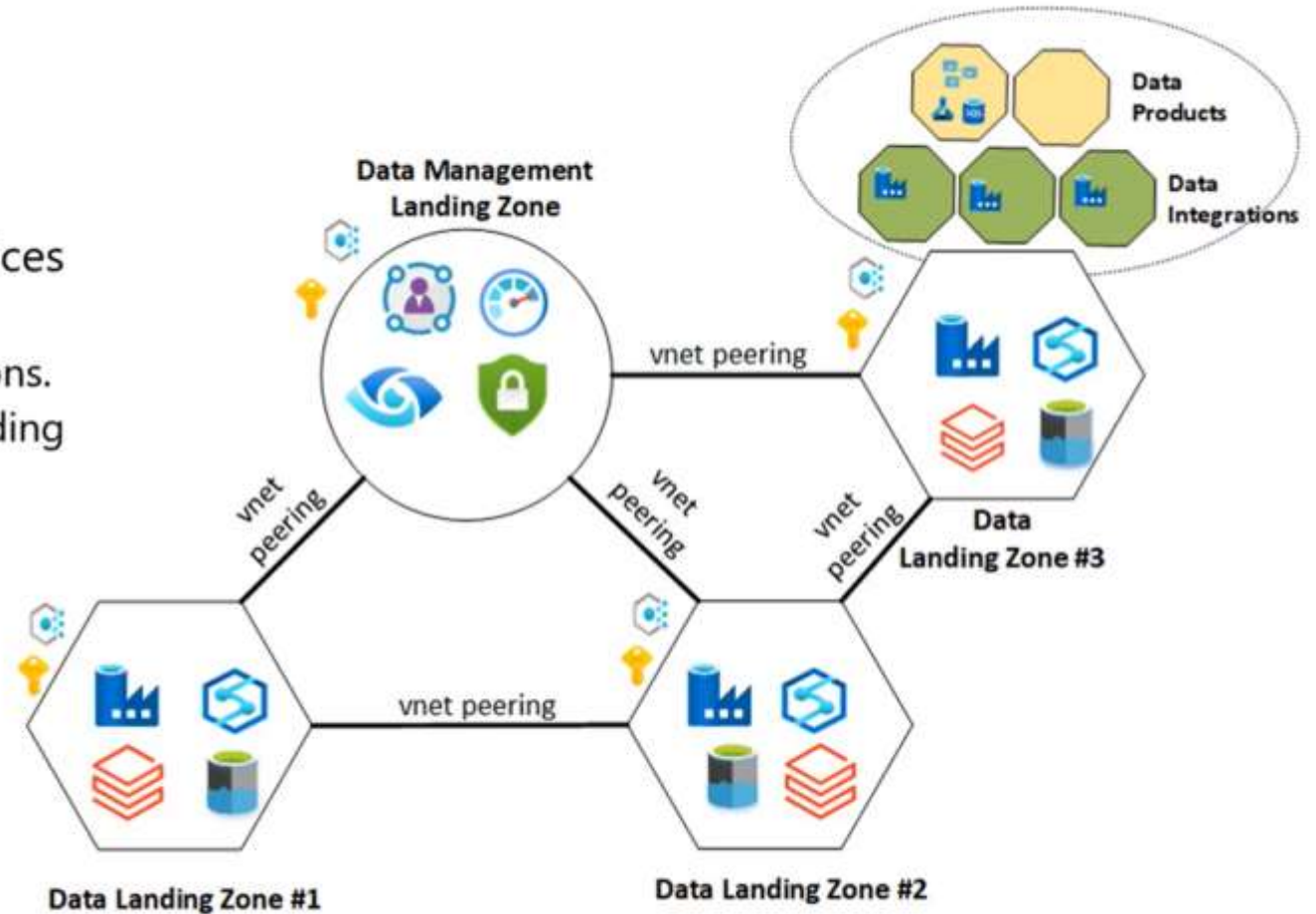
# Single Data Landing Zone Option

- Central data office and no business data offices
- Enable self-service reporting



# Multiple Data Landing Zones Option

- Multiple Data Landing Zone Subscriptions
- Business driven
- Combination of central and business data offices
- Example:
  - Car Group Holding company with multiple divisions.
  - Each car brand would have their own Data Landing Zone
  - One Data Management Landing Zone





# Value Model

#	Description	Value accrued	Customer role alignment
1	A pre-defined architectural model	<ul style="list-style-type: none"> <li>Reduced effort, timescale &amp; risk associated with "starting from scratch" / "custom design efforts"</li> </ul>	<ul style="list-style-type: none"> <li>Implementation Project Manager</li> <li>CTO / Customer Architectural Owners</li> </ul>
2	A pre-defined implementation model	<ul style="list-style-type: none"> <li>Reduced security effort.</li> </ul>	<ul style="list-style-type: none"> <li>CISO / Security teams</li> <li>CTO / Customer Architectural Owners</li> </ul>
3	Architecture and implementation converge to create a starter default solution pattern	<ul style="list-style-type: none"> <li>Reduced design &amp; engineering effort.</li> <li>Shorter implementation timelines</li> </ul>	<ul style="list-style-type: none"> <li>Implementation Project Manager</li> <li>Technical implementation teams</li> </ul>
4	Solution pattern provided as documentation and Dev Ops automated assets (YAML etc) in Git ready for customer versioning and ownership	<ul style="list-style-type: none"> <li>Ability to automate provisioning.</li> <li>Pre-delivered baseline for modification to meet customer needs.</li> <li>Accelerated move to Infrastructure as Code &amp; Dev Ops methods</li> </ul>	<ul style="list-style-type: none"> <li>Dev Ops teams / Ops Teams</li> <li>Implementation Project Manager</li> </ul>
5	Pre-defined data provider and data consumer usage pattern guidance with team models personas etc	<ul style="list-style-type: none"> <li>Reduced time to adopt modern data methods in organisation</li> </ul>	<ul style="list-style-type: none"> <li>CDO / "Data Driven Org" Champions</li> <li>Data Product Owners</li> <li>Data Engineers / Data Test Teams</li> </ul>
6	Pre-defined operational and management models	<ul style="list-style-type: none"> <li>Reduced time to onboard into production.</li> <li>Reduced effort to establish production maturity</li> </ul>	<ul style="list-style-type: none"> <li>Op's teams / Support teams</li> <li>CTO / Customer Architectural owners</li> </ul>

# Leverage the data management and analytics scenario

Accelerate your journey with high value prescribed guidance, resources, and best practices.



# Sample Implementations

<https://github.com/Azure/data-management-zone>

1. Step 1 Deploy the Centralized Governance Hub which Microsoft calls the Data Management Landing Zone
2. Deploy the Product Nodes which Microsoft calls the Data Landing Zones.
3. Inside the Data Landing Zone(s) you can create one or ore resource groups to hold Batch, Streaming or Analytics data, also deployed via templates.

