BEST PRACTICES AWS **SPARX S2E2**

DATA MESH
PRINCIPLES, BENEFIT AND IMPLEMENTATION

by Alexander Hofstetter

IN SHORT

- **When implementing a data mesh, data is viewed as a product, its ownership is transferred to data domains, there is a central self-service infrastructure and governance is mostly the responsibility of the data domains.**
- **This structure improves both the quality and discoverability of data.**
- **AWS can help companies establish a central infrastructure through which data domains can make their data available and grant access to it.**

Watch episode
trivadis.com/sparx

Currently, it is hard to avoid the topic of data mesh. It represents a paradigm shift in the way we think about building data platforms, organising data as a data-driven company and at the same time solving potential problems with already established technologies – such as those used for a central data lake.

The topic was established by Zhamak Dehghani[1], Technology Principal at ThoughtWorks, who also describes the four design principles of a data mesh.

[1] https://martinfowler.com/articles/data-mesh-principles.html
https://martinfowler.com/articles/data-monolith-to-mesh.html

WHAT ARE THE DESIGN PRINCIPLES?

There are four different pillars for a data mesh:

- Domain-Driven Data Ownership Architecture
- Data as a Product
- Self-Service Infrastructure as a Platform
- Federated Computational Governance

DOMAIN-DRIVEN DATA OWNERSHIP ARCHITECTURE

The Domain-Driven Data Ownership Architecture is about moving from centralised data platforms with centralised data ownership to an approach in which ownership of the data is transferred to so-called data domains – comparable to application design.

A central data platform represents a large monolith that covers all areas of the application. In turn, many small microservices have been implemented, each covering only a specific function. In the data mesh, the data domain is the small microservice and the central data platform (e.g. a data lake or an enterprise data warehouse) is the monolith.

The data domains consist of team members who also have the technical expertise about the data. This allows them to better determine which data is collected, how it is transformed or aggregated and how it is best processed. In the data mesh, it is also possible that the data producer is at the same time the consumer of their own data. In other words, the data is distributed at the source and does not have to go via a central unit. The key is to view data as a product.

> **« The key is to view data as a product.»**

When consumers need data, they do not go to central BI/data analytics or data platform teams with their requirements as before, but directly to the data owners. The advantage of this is that the central platform teams are relieved of the burden of setting up new transformation or provision processes. Furthermore, the consumer can communicate directly with the producer, i.e. the entity that knows the data best. To prevent this from creating new data silos, the data domains also get the responsibility to distribute their data products as well as to document them in detail and uniformly.

DATA AS A PRODUCT

Within the data domains, a new role is created, namely that of data product owner. This person is responsible for delivering the data as a product. It is important to describe the data precisely so that data analysts as well as other data domains can find it. In addition, the data product owner is responsible for the quality and growth of the data. The data product owner knows how many consumers use their data and how satisfied they are.

SELF-SERVICE INFRASTRUCTURE AS A PLATFORM

The decentralised data owners need a central infrastructure as a platform for their data. In this context, it is particularly important that the infrastructure remains domain-agnostic and that no responsibility is transferred to a central component. This means that the teams use the data platform and by doing so can control themselves what data they make available and who can access it.

Accordingly ,with Self-Service Infrastructure as a Platform, only a self-service is made available to use this central infrastructure. This has the advantage that the data domains use the same standards for the infrastructure without having to work with a central unit beforehand. This relieves the central unit.

FEDERATED COMPUTATIONAL GOVERNANCE

The last principle ensures that the different data products can be used sensibly. Data Mesh basically follows a distributed approach – Federated Computational Governance takes this into account by not setting up and living governance exclusively centrally, but delegating a lot of responsibility to the data domains. Decisions that affect the inner workings of a domain should also be made by the domain. Nevertheless, there is also a need for central governance that deals with topics that are of overarching interest, such as standardisation of interfaces, use of a data catalogue, as well as the management of access in compliance with the domain›s own compliance and security rules.

> **« In terms of governance, a lot of responsibility is delegated to the data domains: Decisions that affect the inner workings of a domain should also be made by the domain.»**

IS DATA MESH A SOLUTION FOR ME?

In addition to the many added values – e.g. data ownership and accountability and the possibility of linking business requirements to the solution and scaling it flexibly to organisational structures – implementing a data mesh also poses certain challenges.

For example, a data mesh requires distributed knowledge and skills in the data domains. There is also the risk of divergent technology stacks. Product thinking and the creation of data as a product is also a complex process that must be supported by all employees involved.

The scope of establishing a data mesh can be compared to the switch to DevOps – it is not just about implementing a new data platform.

Therefore, it is essential to check which hurdles have to be overcome in order to use a data mesh. Questions that arise are:

- How can I break apart existing data platforms and integrate them into a data mesh?
- Can I establish the mindset of seeing data as a product in the company?
- Do I have employees who can act as data product owners and document, evaluate and market the data? And can also be held responsible for it?

Ultimately, the implementation process also depends on whether you already have data platforms in operation or are just starting them up.

HOW CAN AWS HELP ME WITH THIS?

While you have to manage the shift to data-driven domains and treating data as a product on your own, AWS has tools that make it possible to deploy infrastructure as a platform while maintaining governance, compliance and security aspects. Among the most important is a managed data lake service called AWS Lake Formation.

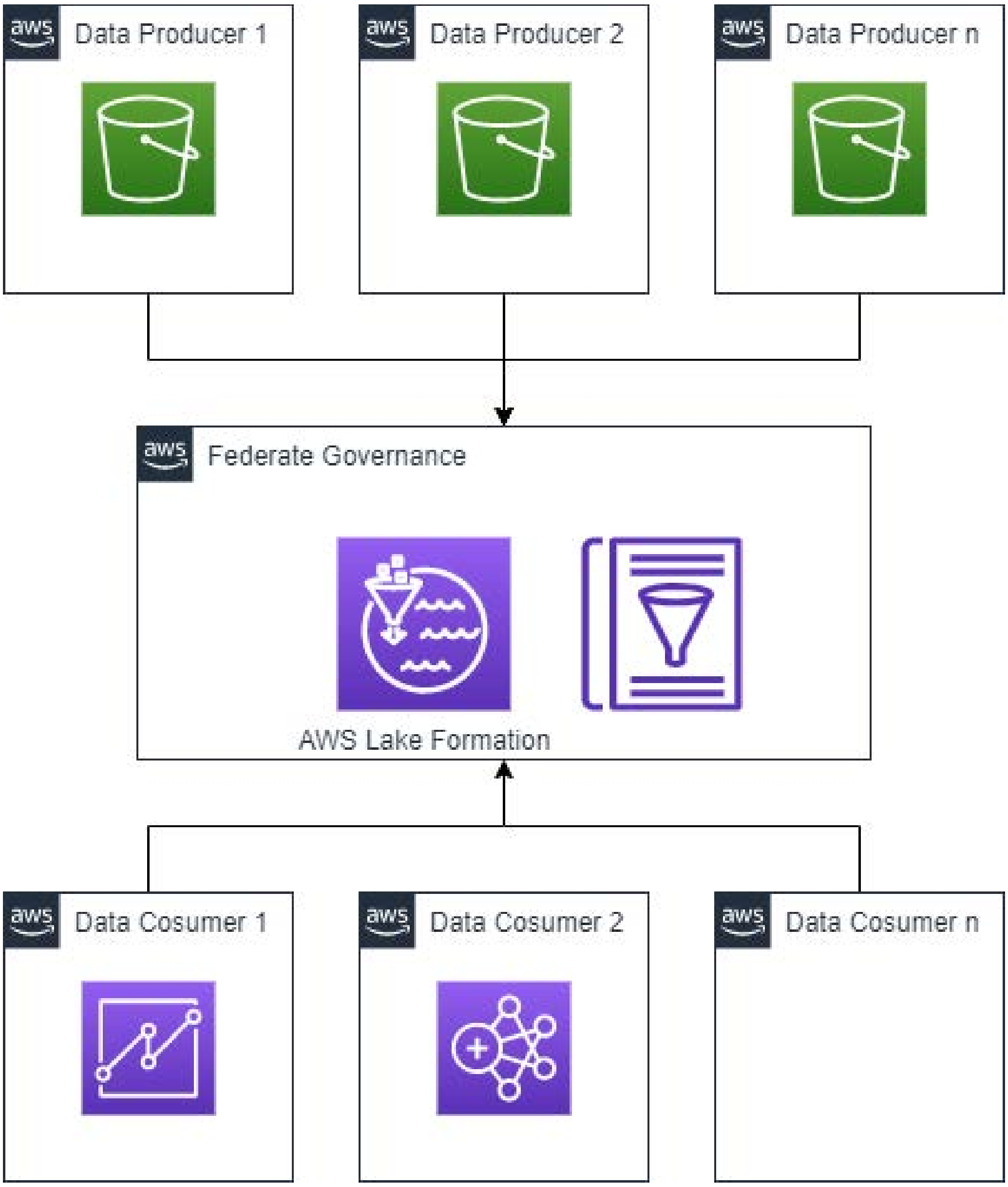
With AWS Lake Formation, data domains can easily migrate, transform, cleanse and catalogue their data into a data lake and make it available to data consumers.

AWS Lake Formation also provides a unified authorisation layer to ensure access to the data. Another important aspect is that the data does not have to be copied to the consumer first, but can be accessed directly.

This provides a central platform for the data domains, which can make their own data available to any consumer and control their access. Access takes place via APIs and authorisation via a security token.

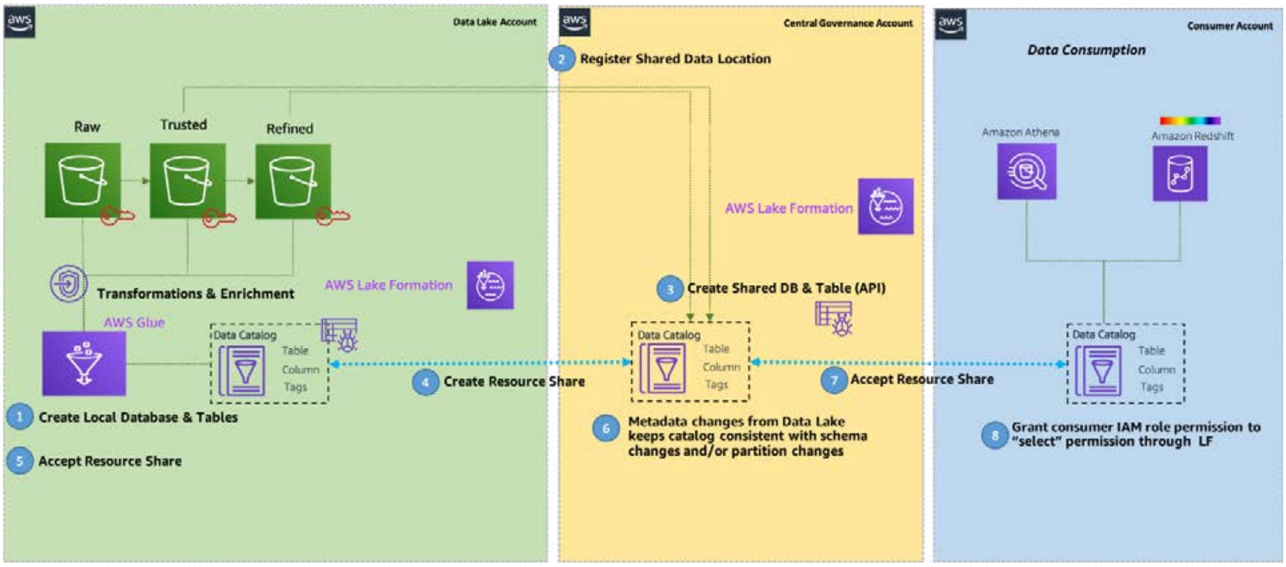
WHAT DOES AN AWS LAKE FORMATION ARCHITECTURE LOOK LIKE?

Data producers register their data in AWS Lake Formation. Consumers can then request access to the data and use it directly if it is approved.



HOW IS THE ACCESS DESIGNED?

Below is a workflow from the registration of the data by the data producer to the release and use by the consumer:


aws
Data Lake Account
Raw
Trusted
Refined
Transformations & Enrichment
AWS Lake Formation
AWS Glue
Data Catalog
Table
Column
Tags
1 Create Local Database & Tables
5 Accept Resource Share
4 Create Resource Share
aws
Central Governance Account
2 Register Shared Data Location
AWS Lake Formation
3 Create Shared DB & Table (API)
Data Catalog
Table
Column
Tags
6 Metadata changes from Data Lake keeps catalog consistent with schema changes and/or partition changes
7 Accept Resource Share
aws
Consumer Account
Data Consumption
Amazon Athena
Amazon Redshift
Data Catalog
Table
Column
Tags
8 Grant consumer IAM role permission to "select" permission through LF

Source: https://aws.amazon.com/blogs/big-data/design-a-data-mesh-architecture-using-aws-lake-formation-and-aws-glue/

CONCLUSION

Implementing a data mesh goes far beyond technology and can bring many benefits. For the technical implementation, AWS provides tools that allow data domains to use a simple and secure environment to provide their data as well as grant access to it. This keeps the responsibility in the teams and allows data consumers to talk directly to the owners of the data.

The newly created role of the data product owner improves the quality of the data. Thanks to good documentation, it is also easier for potential data consumers to find.

ABOUT THE AUTHOR

Alexander Hofstetter

Alexander Hofstetter is a Principal Consultant at Trivadis - Part of Accenture in Munich in the field of Infrastructure Managed Services. One of his focal points is Oracle database administration. In addition to automation, Hofstetter has been working on the administration of Big Data clusters since 2017. In 2018, the design of AWS Cloud architectures and the automated construction of infrastructures in the AWS Cloud with Infrastructure as Code became another focus. He currently works as a product owner in the AWS area at the Cloud Competence Center.

EMAIL alexander.hofstetter@trivadis.com

PHONE +49 89 9927 59 302