Starburst
ANALYTICS ANYWHERE

E-BOOK

Data Mesh 101: What It Is & Why You Need It

Data Mesh, a new architectural approach founded by Zhamak Dehghani, is a holistic paradigm that sees datasets as data products, owned and managed by domains. The idea is that each domain contains business experts and its own embedded data engineers. Data Product owners can manage that data with other domains and end consumers, while driving a level of data ownership and responsibility. This ownership is often lacking in current data platforms that are largely based around centralized, monolithic, and complex pipelines.

Data Mesh can be thought of as an approach towards decentralizing ownership, transformation, and serving of data. But what does that mean?

Today's data landscapes are dominated by an anti-pattern that segments the architecture into three areas, each having its own concerns and sets of responsibilities. These teams also have little to no responsibility for the other areas.

Zhamak Dehghani
Founder of the
Data Mesh Paradigm

> “A decentralized sociotechnical approach in managing and accessing analytical data at scale.

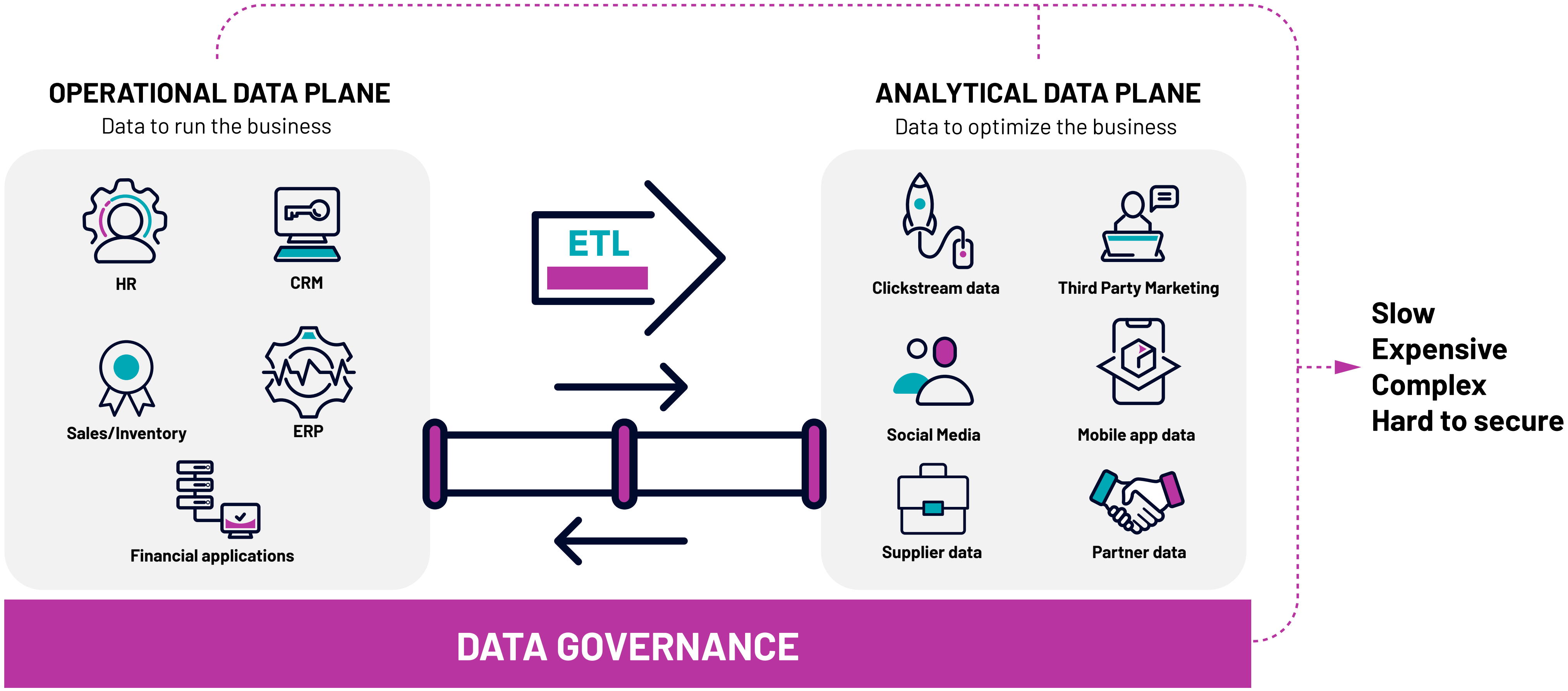
Today’s Data Architecture Landscape

1. The **operational data plane** is where data is generated and runs the business. This plane typically consists of a large number of technologies, business processes, and is supported by teams.

2. The **analytical data plane** is where data is stored and transformed to meet the needs of *optimizing* the business. This optimization could take the form of descriptive, diagnostic, or predictive analytics. Traditionally, this is where data warehouse and data lake technologies reside, and the teams that support them.

3. **ETL/ELT or data pipelines** nest between these planes, bringing data from the operational to the analytical plane. The teams here are responsible for an awful lot. They have to understand the data models and technologies used behind operational applications, the technologies and techniques involved in transforming data, and the analytical plane data models and technologies. The cognitive load of the individuals here is significant.

Figure 1: Today’s Data Architecture Landscape



Arguably this structure has been ripe for change for a very long time. This was present in the early days of on-premise warehousing, the adoption of data lakes, and now in the era of cloud data warehouses. Continuing to do the same and expecting different results seems like the proverbial definition of madness. As data usage gets more ubiquitous, the hindrance caused by the lack of visibility between these planes can lead to missed revenue opportunities, increased costs and greater exposure to organizational risks.

Data engineers who own the ETL process have the most visibility to what happens in each plane. Except, with all the heavy lifting to move that data, understand operational and analytical data models, and the underlying technology, it becomes an immense challenge to quickly respond to internal and external business changes. This separation of concerns results in the symptoms below:

Symptoms of Misfiring Data Programs

Inability to scale data sources

Inability to scale data users

Inability to effectively bootstrap

Inability to materialize value from data

The most common challenge is an inability to add new data sources in the operational plane and for data to flow to the analytical plane in a timely manner. Furthermore, an ***inability*** to support users to access data in new and different ways, as well as supporting new data initiatives will reduce the date value that we can materialize from our investment.

With Data Mesh as a significant organizational change, its objectives are to:

- enable an organization to be agile
- respond to internal and external change and opportunity over the long term
- sustain a competitive advantage, *and*
- support a proportionate return on the investment made in data and analytics

Data Mesh removes bottlenecks between operational and analytical systems. The connection between the planes will be supported and owned by business domains which include business skills, technical skills, and data skills. Fragile pipelines that hold up innovation and change are replaced by a modularized pipeline, where the modules are owned by the domains and malleable to meet business needs. Lastly, Data Mesh has answers for the lack of readily available data engineering talent; the self-service platform abstraction layer allows generalists to free up and augment data engineering talent to focus on higher-value tasks.

DATA MESH OBJECTIVES

Respond gracefully to business' essential complexity, volatility, and uncertainty

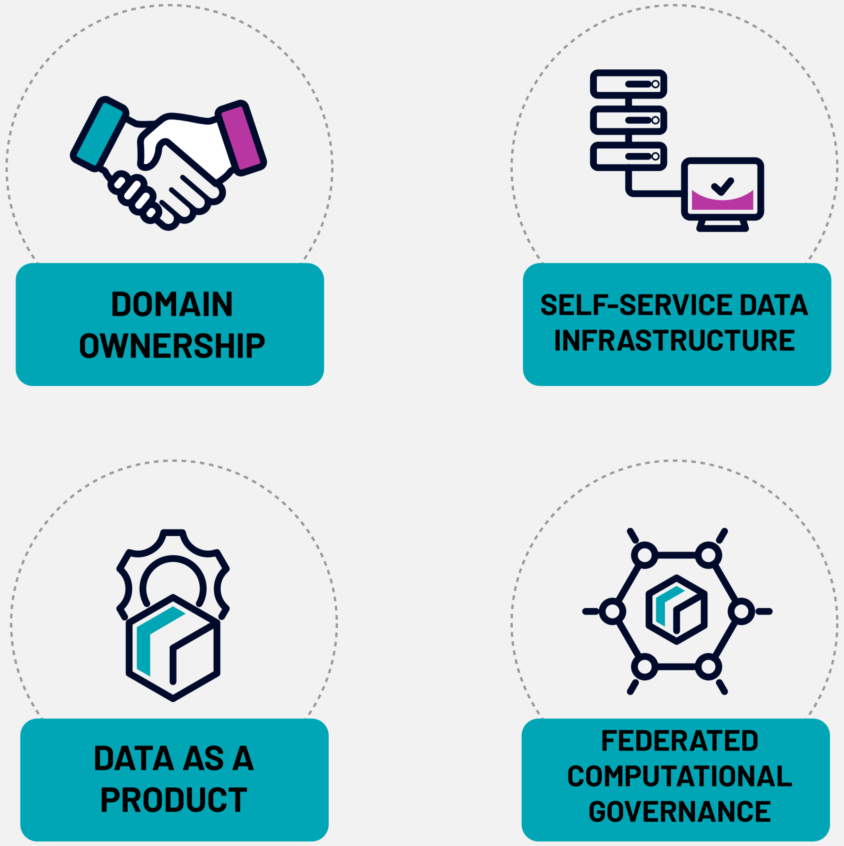
Sustain agility in the face of growth

Accelerate getting value proportionate to the investment

PAST BLOCKERS	DATA MESH ENABLERS
Monolithic centralized bottleneck	Align business domain, tech and data
Data pipeline fragility and coordination	Remove pipeline and localize code and data into one unit with clear contract
Lack of talent - data engineers - to manage data	New data platform **abstraction** to allow generalists to develop and use data

Data Mesh is Founded on Four Principles:

The Principles of Data Mesh



DOMAIN OWNERSHIP

SELF-SERVICE DATA INFRASTRUCTURE

DATA AS A PRODUCT

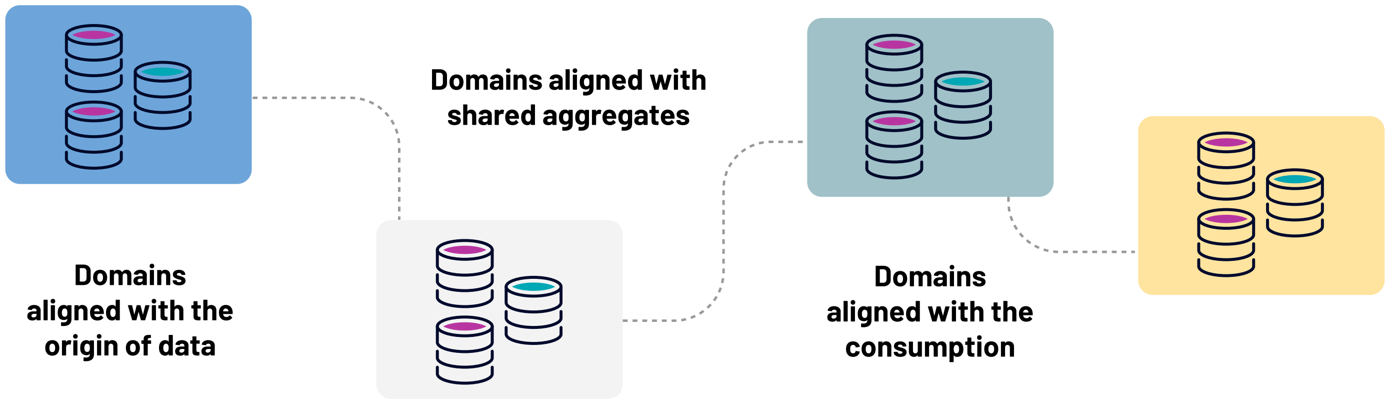
FEDERATED COMPUTATIONAL GOVERNANCE

Let's get into what each of these means.

Domain Ownership

A "domain" is an organizational structure where users have knowledge and ownership of operational data, the data pipeline, and analytical data. This influences the way the end-user will consume that data, related to a specific set of data. And thus a team of individuals who master that process is brought together within the domain to execute the process from start to finish. How those domains are defined is dependent on the needs of the business. Some domains are related to customer data, or product, while others could be around finance or operations. One approach is to align domains with the source of the data, whilst another is to align the domains with the consumption of the data. However, it's likely that the best approach is to align domains to existing business functions.

Figure 2: Distributed Data Domain Ownership


Domains aligned with shared aggregates
Domains aligned with the origin of data
Domains aligned with the consumption

This is very different from the landscape organization seen earlier. So how are these domains motivated to take on all that responsibility? It takes a shift in culture, which is why Data Mesh is referred to as more than just an architecture shift, but rather a sociotechnical approach. Making this cultural change is the subject of another article.

If we consider the domains in an e-commerce site, we might have domains focused around the data collected about merchants, users, marketing, and best sellers. The visual below shows that the data products, owned by the domains and their usage across domains *form* the 'Mesh'. The domains or data products are the nodes and the use of the data products across domains are the relationships or 'edges' between the nodes.

Figure 3: An Example of Domains - Ecommerce Site

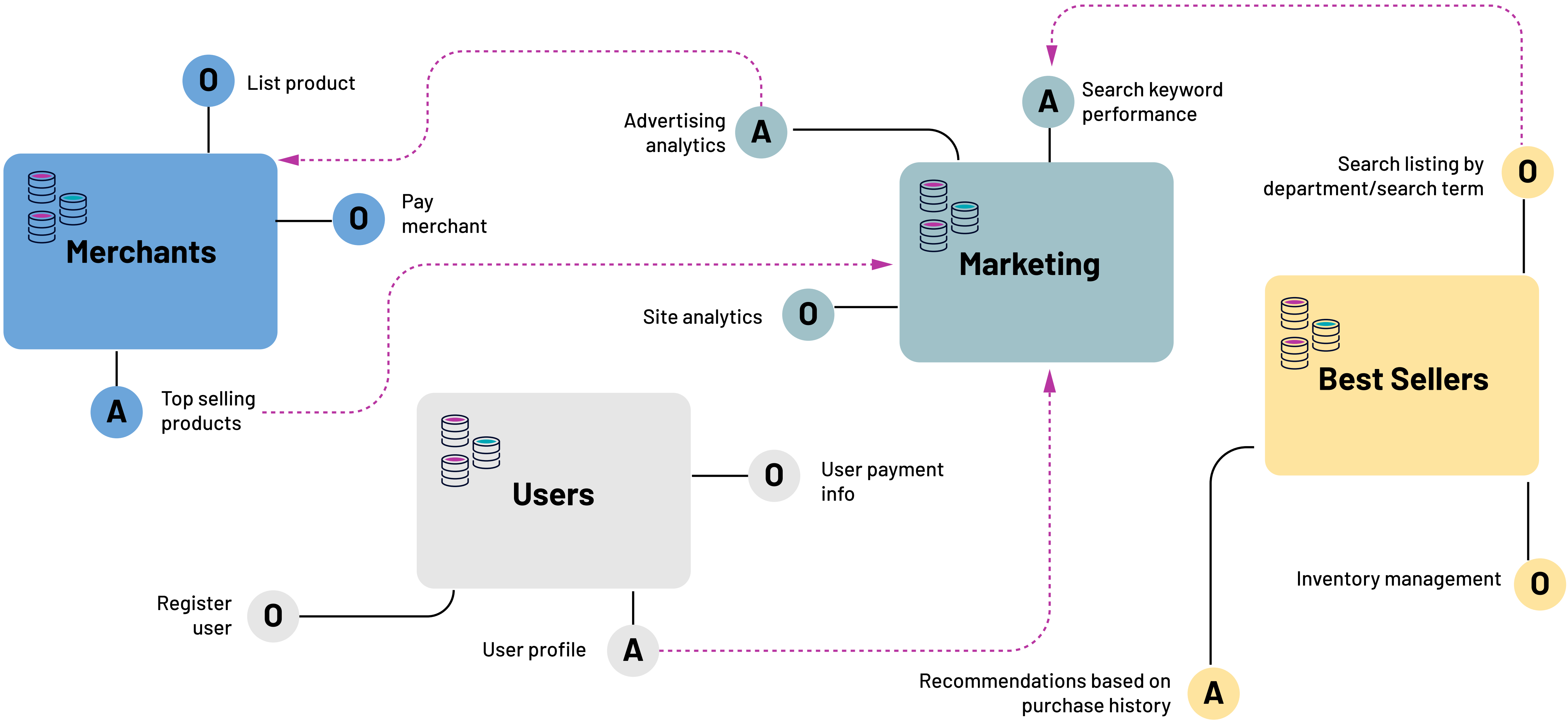
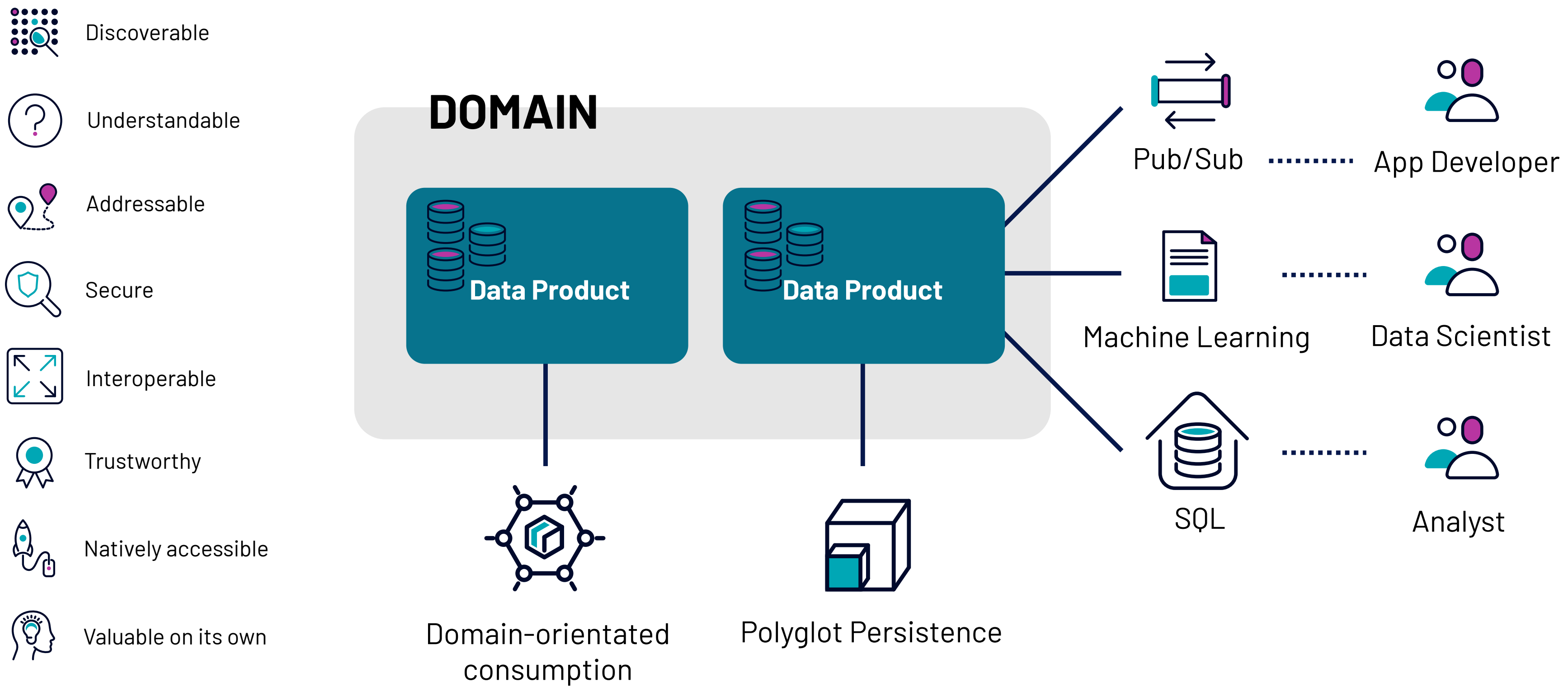


Figure 4: Baseline Usability



Data as a Product

The principle of treating data as a product is the heart of Data Mesh. It's called a "product" because using it brings value to its consumer. Essentially, the use of a data product can be a proxy for value. If consumers do not use the data product, then the product can be considered of lower value. Alternatively, if the data product is heavily used, then its value can be considered to be higher.

Baseline Usability

A data product can be thought of as an artifact of a domain with two types of ports:

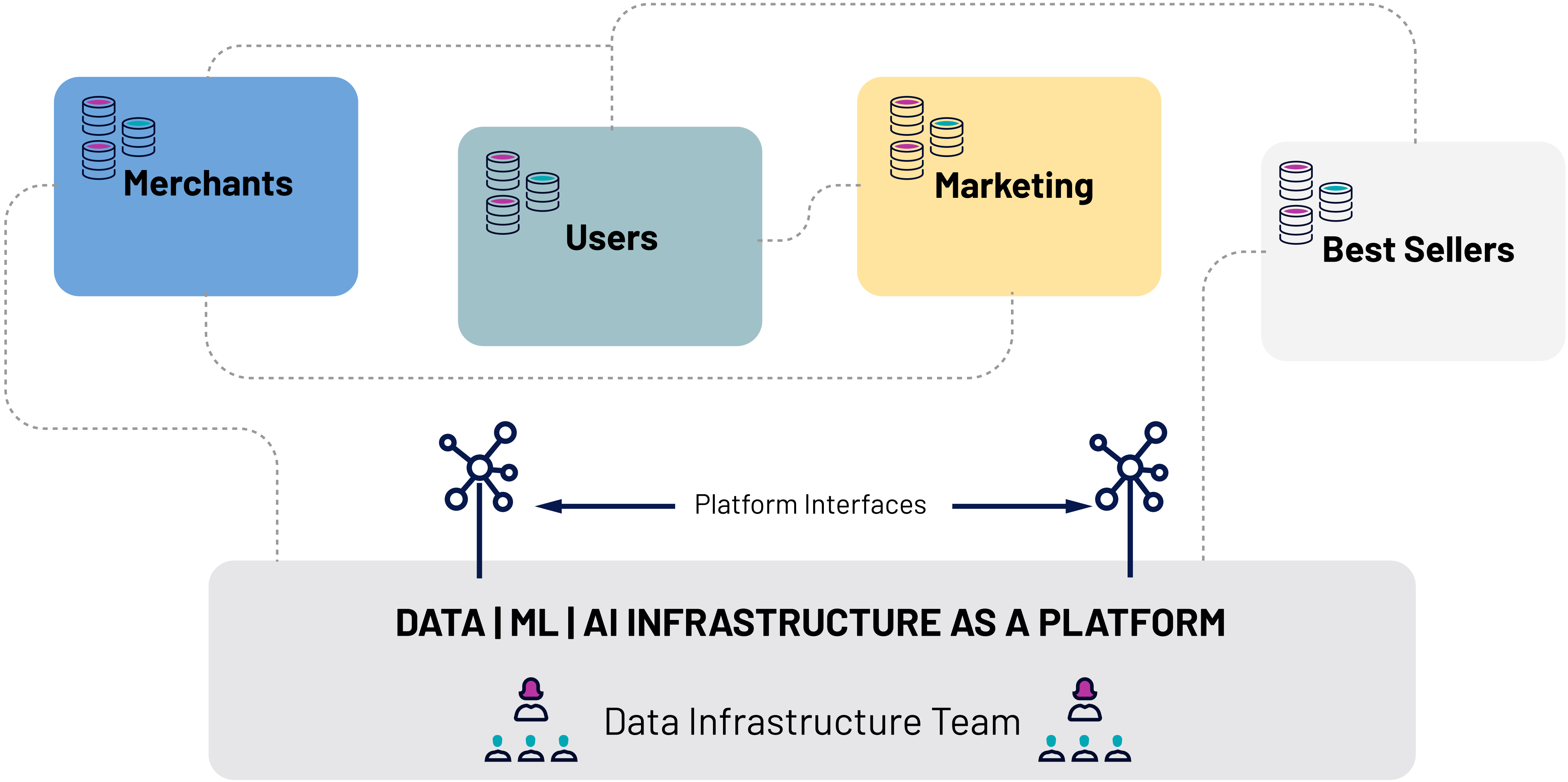
- Input ports that are either operational systems or data products from other domains, and
- Output ports, which make data available externally to downstream consumers.

A simple definition of a data product is a data artifact that is exposed externally for consumption. Data products have a number of capabilities and characteristics that include metadata, which enable end-users to find and use data products in a secure and governed manner. For domains to create data products, we need technology that the members of the domain can readily use.

Self-Service Infrastructure

The infrastructure platform of a Data Mesh is considered to be separate from the data. Historically, those would have been owned by the same team or person if we think of a DBA, for example. The data is the responsibility of the domains, while the infrastructure is solely owned by the platform and team that operates it.

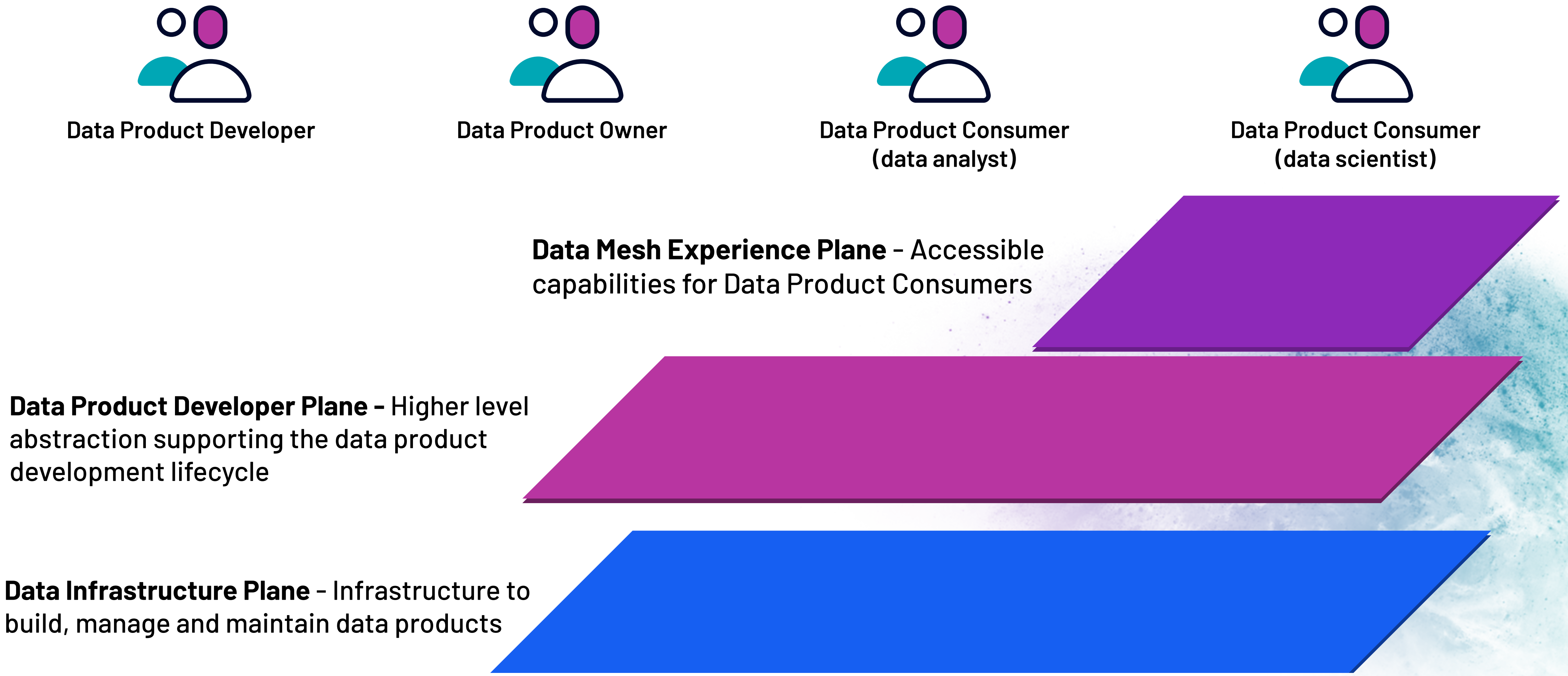
Figure 5: Enable Autonomy: Self-Service Infrastructure



Data product developers who work and occupy the domain should have no knowledge of the technicalities of the infrastructure. Rather, they work in higher-level abstraction layers that allow them to utilize the infrastructure, without having to consider the underlying infrastructure or other technology, such as where the data physically resides.

Domains that use a self-service infrastructure to create data products could lead to chaos if those data products were built to different standards and used different definitions for data entities. This is where governance has a significant role to play.

Figure 6: Logical Architecture



Data Product Developer

Data Product Owner

Data Product Consumer (data analyst)

Data Product Consumer (data scientist)

Data Mesh Experience Plane - Accessible capabilities for Data Product Consumers

Data Product Developer Plane - Higher level abstraction supporting the data product development lifecycle

Data Infrastructure Plane - Infrastructure to build, manage and maintain data products

Federated Computational Governance

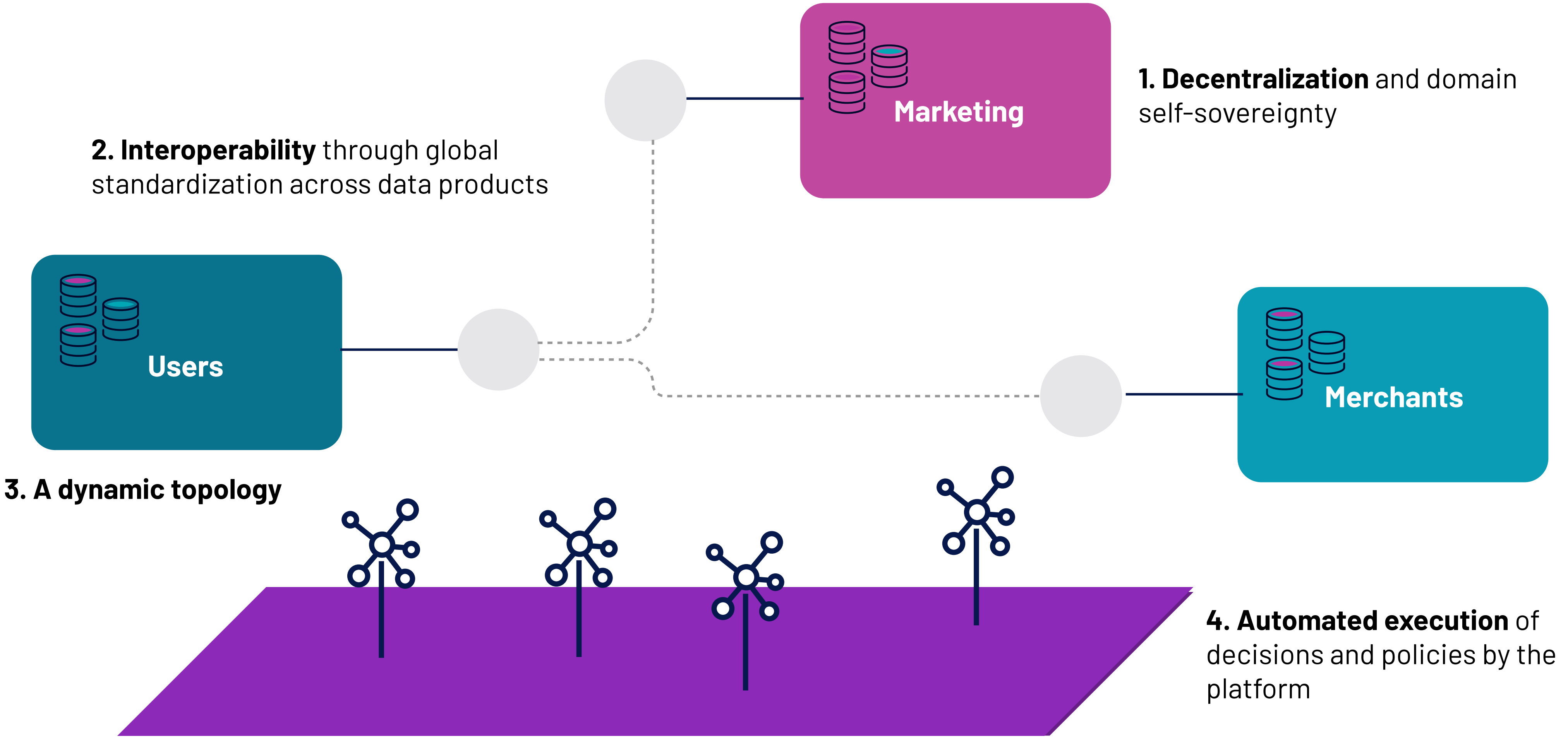
Data Mesh is a fairly new ideology for most organizations, and those who are curious are often concerned with the first two principles: domain ownership and data as a product. For that reason, there hasn't been as much focus on governance to this point, but one can certainly conceptualize it based on the framework proposed by Zhamak Dehghani in her upcoming O'Reilly book, Data Mesh: Delivering Data-Driven Value at Scale.

To support decentralized domains, we need two levels of governance: at the domain level and at the mesh level. Domains are responsible for owning governance within the domain, and ideally contributing to a governance policy definition at the mesh level. This brings us to a federated governance scenario. To define the policies, we could bring together domain owners and members of the infrastructure platform team.

They'll need to ask questions like:

- What metadata does a data product have to make available?
- What standards do they have to conform to?
- What SLAs do we want?
- Do we want SLAs defined within the domains, maybe at the data product level?

Figure 7: Federated Governance

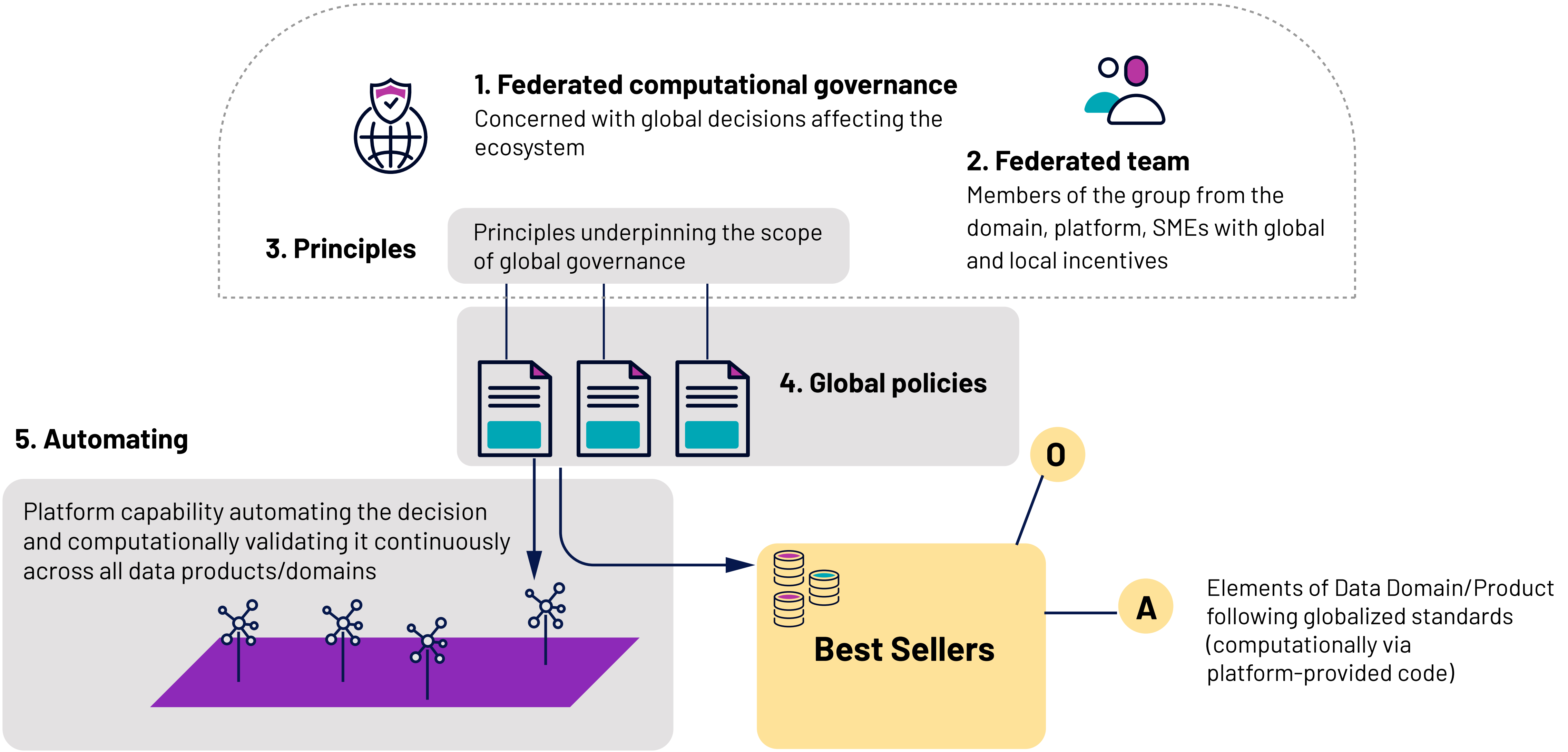

Marketing
1. **Decentralization** and domain self-sovereignty
2. **Interoperability** through global standardization across data products
Users
Merchants
3. **A dynamic topology**
4. **Automated execution** of decisions and policies by the platform

And so these policies need to be defined, both at a domain and mesh level. Then, those policies need to be adopted and implemented within the self--service platform so that when data products are published to the Data Mesh, they conform to these policies in an automated way.

The idea that domains can be responsible for data from ingestion to serving is a very powerful one. The idea that data products exist that add immense value to the organization is *equally* powerful.

A possible outcome could be the emergence of data silos, with data that is so decentralized that it can't be used together. Fortunately, this federated governance approach prevents this from happening. This can also be prevented through the enforcement of policies and standards that all data products must follow. This all works within the infrastructure platform to create a Data Mesh, which will provide a great deal of value for the organizations that adopt it.

Figure 8: Federated Governance


1. Federated computational governance
Concerned with global decisions affecting the ecosystem
2. Federated team
Members of the group from the domain, platform, SMEs with global and local incentives
3. Principles
Principles underpinning the scope of global governance
4. Global policies
5. Automating
Platform capability automating the decision and computationally validating it continuously across all data products/domains
Best Sellers
O
A
Elements of Data Domain/Product following globalized standards (computationally via platform-provided code)

To summarize, Data Mesh is about a holistic organizational, architectural, and technological approach to get the most out of your data team and your data. Data Mesh is not one type of technology or code that magically solves data problems at the touch of a button. Instead, it's about rethinking the *human side* of technology, introducing new enabling technologies and adopting a more open approach to building data platforms at scale.

To learn more about how you can implement a Data Mesh at your organization, visit our Data Mesh Resource Center now.

How Starburst Can Help

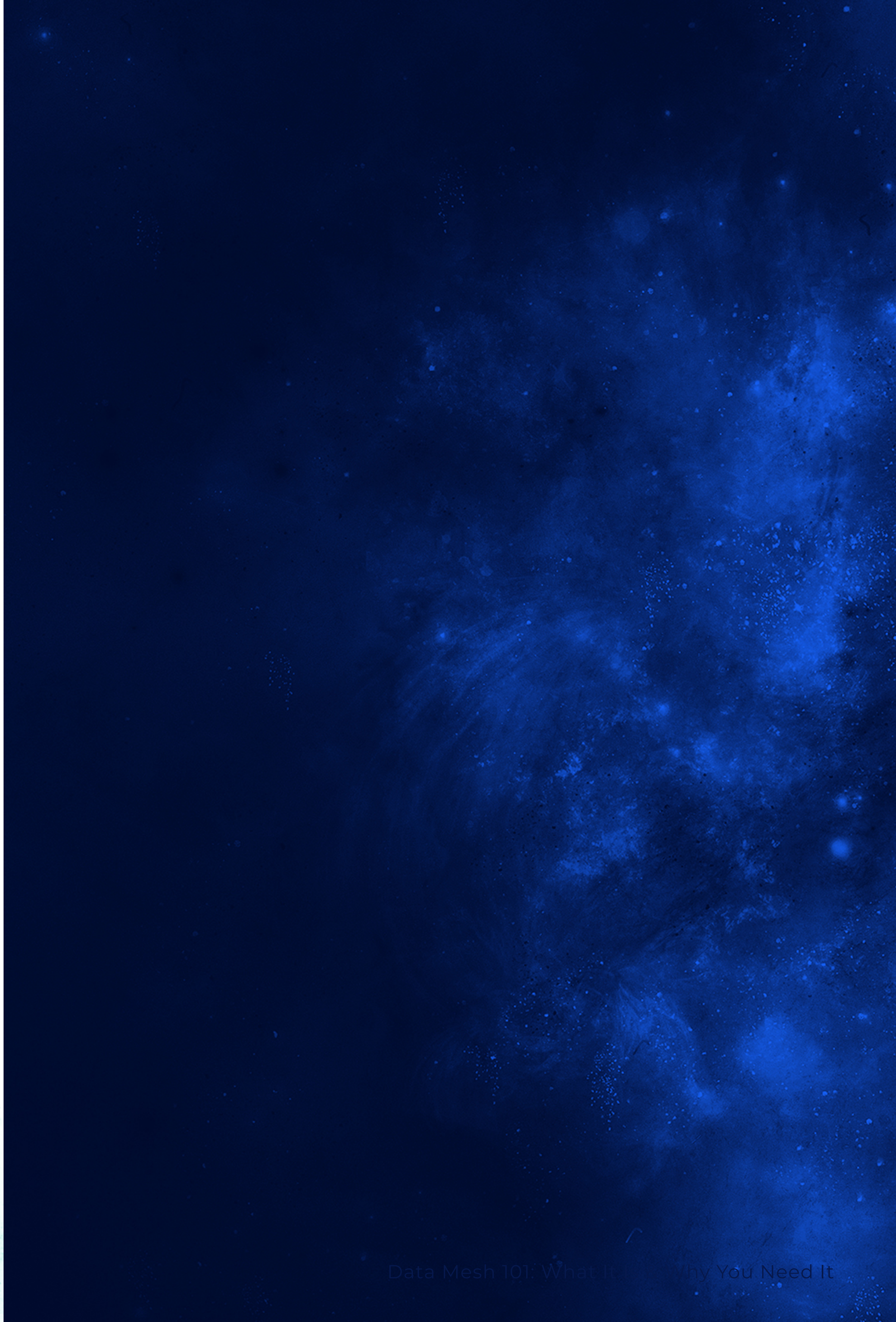
Companies adopting a Data Mesh architecture must have an analytics engine capable of federating across these different data sources. Starburst is the analytics engine for the Data Mesh architecture, providing a single point of access to distributed data and empowering self-service analytics for each of the business domains.

With Starburst, there's no need to chase the idea of a single source of truth. Data is maintained by the domain owners but easily accessible in real-time across your organization. Starburst is built on open-source Trino, a distributed engine that can execute SQL queries against data stored in a range of databases and file systems. With Starburst and Trino, teams can lower the total cost of their infrastructure and analytics investments, prevent vendor lock-in, and use the existing tools that work for their business so that they can concentrate on enabling faster time-to-insights. Trino's open technology means that integration with other open technologies such as data catalogs and data discovery tools is simpler and reduces the total cost of ownership of the self-service data platform.

If you are moving towards adopting a Data Mesh architecture, we want to be there to help. Visit our Data Mesh Resource Center for more information.

Contact Us

www.starburst.io/contact/



Starburst

ANALYTICS ANYWHERE

STARBURST.IO