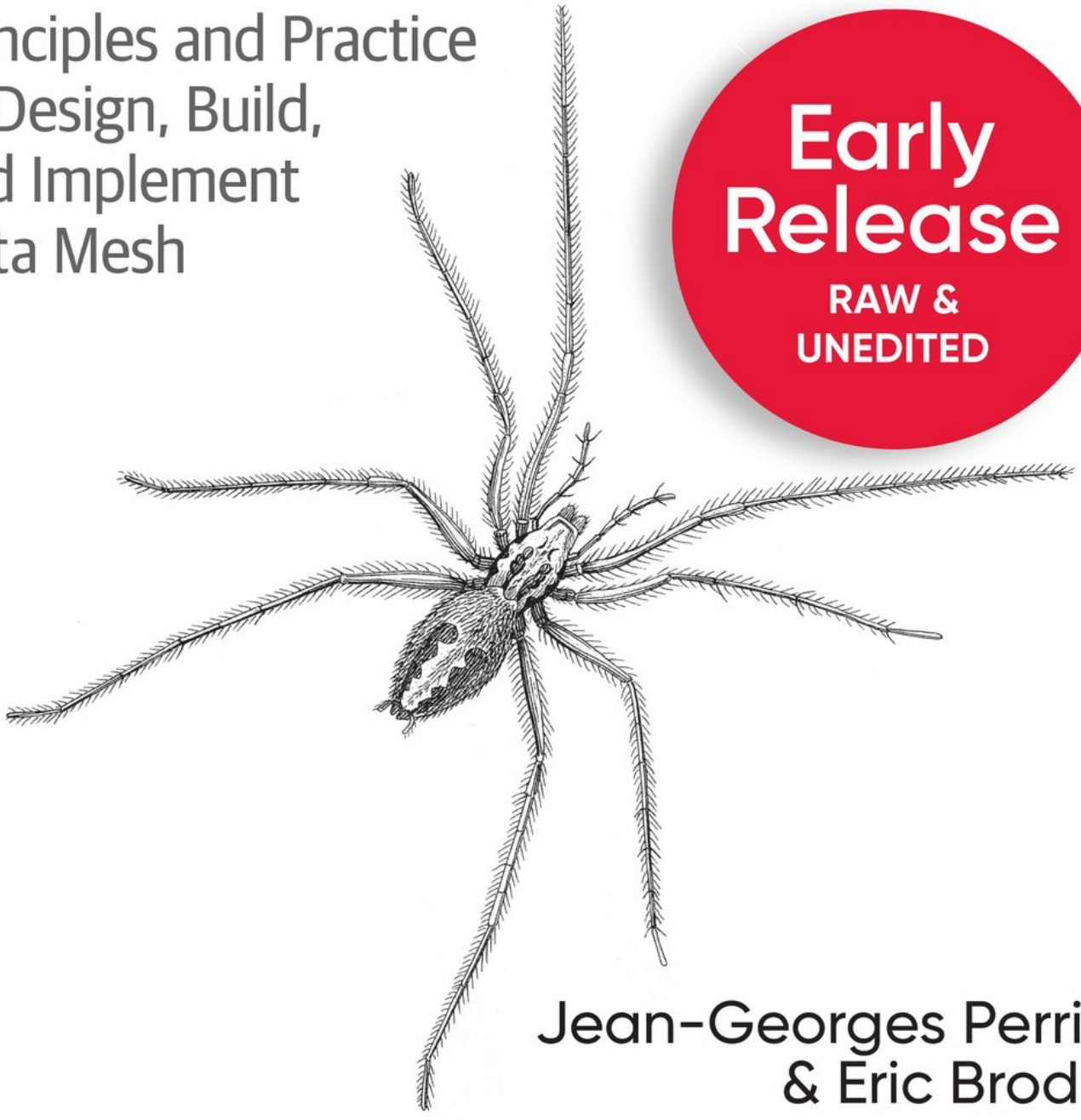
O'REILLY®

Implementing Data Mesh

Principles and Practice to Design, Build, and Implement Data Mesh

Early Release
RAW & UNEDITED



Jean-Georges Perrin
& Eric Broda

Implementing Data Mesh

Principles and Practice to Design, Build, and Implement Data Mesh

With Early Release ebooks, you get books in their earliest form—the author's raw and unedited content as they write—so you can take advantage of these technologies long before the official release of these titles.

Jean-Georges Perrin and Eric Broda

Beijing • Boston • Farnham • Sebastopol • Tokyo

Implementing Data Mesh

by JG Perrin and Eric Broda

Printed in the United States of America.

Published by O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472.

O'Reilly books may be purchased for educational, business, or sales promotional use. Online editions are also available for most titles (http://oreilly.com). For more information, contact our corporate/institutional sales department: 800-998-9938 or *corporate@oreilly.com*.

- Editors: Shira Evans and Aaron Black
- Production Editor: Beth Kelly
- Interior Designer: David Futato
- Cover Designer: Karen Montgomery
- Illustrator: Kate Dullea

- December 2024: First Edition

Revision History for the Early Release

- 2023-08-23: First Release
- 2023-11-21: Second Release
- 2023-12-19: Third Release
- 2024-01-24: Fourth Release
- 2024-03-26: Fifth Release
- 2024-05-08: Sixth Release

See http://oreilly.com/catalog/errata.csp?isbn=9781098156220 for release details.

or reliance on this work. Use of the information and instructions contained in this work is at your own risk. If any code samples or other technology this work contains or describes is subject to open source licenses or the intellectual property rights of others, it is your responsibility to ensure that your use thereof complies with such licenses and/or rights.

978-1-09815-616-9

[FILL IN]

Brief Table of Contents (*Not Yet Final*)

Chapter 10: Running and operating your Data Mesh (unavailable)

Chapter 11: Implementing a Data Mesh Marketplace (unavailable)

Chapter 12: Implementing Data Mesh Governance (unavailable)

Chapter 13: Running your Data Mesh Factory (unavailable)

Chapter 14: Defining and Establishing the Data Mesh Team (available)

Chapter 15: Defining an Operating model for Data Mesh (available)

Chapter 16: Establishing a practical Data Mesh roadmap (unavailable)

Part I. The basics

This first part of the book aims at setting the decorum for the rest of the book: at the end of this part, you will be familiar with our terminology and our use case.

Chapter 1. Understanding Data Mesh - The Essentials

A NOTE FOR EARLY RELEASE READERS

With Early Release ebooks, you get books in their earliest form—the author's raw and unedited content as they write—so you can take advantage of these technologies long before the official release of these titles.

This will be the 1st chapter of the final book. Please note that the GitHub repo will be made active later on.

If you have comments about how we might improve the content and/or examples in this book, or if you notice missing material within this chapter, please reach out to the editor at sevans@gmail.com.

In the rapidly evolving landscape of enterprise data management, Data Mesh has evolved from being an emerging concept into a cornerstone of modern data architecture. Its ascent marks a significant shift in how organizations handle the ever-increasing complexity and scale of their data ecosystems. The foundational principles of Data Mesh, articulated in

Zhamak Dehghani's seminal work, "Data Mesh: Delivering Data-Driven Value at Scale," have set the stage for a new era in data handling and utilization.

Building upon Dehghani's principles, this book aims to bridge the gap between theoretical understanding and practical application, turning the principles of Data Mesh into practice for data professionals. Recognizing that many of our readers are likely familiar with Dehghani's principles, we delve deeper, not just reiterating these concepts but expanding upon them to demonstrate their implementation in real-world scenarios.

For those new to Data Mesh, we provide an accessible introduction, ensuring all readers are on the same footing. Our book is anchored in the core principles of Data Mesh but extends well beyond this solid foundation to illustrate how these principles can be effectively implemented and operationalized within your organization.

Let's start by reiterating Dehghani's transformative vision, which rests on several key principles:

Decentralized Domain Ownership:

Responsibility for data is distributed among domain-specific teams, each accountable for the quality,

accessibility, and governance of their data.

Data as a Product:

Data is treated as a valuable product, with domain teams responsible for developing and delivering data solutions tailored to their specific needs.

Self-Serve Data Infrastructure:

A framework that empowers domain teams to independently manage their data, reducing dependence on centralized data teams.

Federated Computational Governance:

A model where domain teams enforce data governance within their purview, aligned with overarching organizational policies.

These principles echo the spirit of the Agile methodology in software development. The Agile Manifesto, published in 2001 is still a pivotal document in the software industry which at its core emphasizes individuals and interactions, working software, customer collaboration, and responding to change. These principles were translated into practices through frameworks like Scrum and Kanban, which promote iterative

development, regular feedback loops, and close collaboration between cross-functional teams.

Now, since Agile Manifesto was published, there have been over twenty years of turning core agile principles into practice. We now deliver software faster, better, and cheaper: McKinsey, a consulting firm, has shown that “agile organizations have a 70 percent chance of being in the top quartile of organizational health, the best indicator of long-term performance.” Simply put, the software engineering world has never been the same.

Similarly, Data Mesh introduces agility into the data landscape, emphasizing decentralized ownership, responsive data management, and collaborative cross-functional teams. Just as Agile promotes self-organizing teams, Data Mesh advocates for domain-oriented decentralized ownership, putting the power of data in the hands of individual domain teams. In an Agile context, customer collaboration involves continuous engagement with stakeholders to understand their evolving needs. Likewise, Data Mesh encourages domain teams to engage with data consumers within their organization, gathering feedback, and iterating on their data products to meet their specific requirements.

Just as Agile values working software, Data Mesh places a premium on delivering high-quality data products. Agile-based user stories define the desired functionality, Data Products outline the features, quality requirements, and accessibility of data, enabling domain teams to build and deliver data products that provide real value to their stakeholders.

Adopting Local Autonomy, Speed, and Agility

Data Mesh, in many respects, brings agility to data. And in doing so, Data Mesh offers several benefits that address the challenges organizations face in data management, particularly in relation to adopting local autonomy, speed, and agility.

First, it advocates for local autonomy. Traditional centralized approaches often result in overloaded data teams and bottlenecks in decision-making. In contrast, Data Mesh empowers individual domain teams with the ownership and responsibility for their data. This decentralization allows teams to have a deeper understanding of their specific data needs and requirements, leading to more effective decision-making and faster response times. By fostering local autonomy, Data Mesh enables teams to adapt quickly to changing data demands and

make data-driven decisions in a timely manner. And with local autonomy, Data Mesh enables speed, and with increased speed, faster time-to-market.

With its focus on self-serve data infrastructure, Data Mesh enables domain teams to access and manage their data independently. This eliminates the need for bureaucratic processes and time-consuming requests to centralized data teams, reducing wait times and accelerating the data development lifecycle. By putting the necessary tools and resources into the hands of data practitioners, Data Mesh enables rapid iteration, experimentation, and delivery of data products. This increased speed allows organizations to capitalize on data insights more efficiently, gaining a competitive advantage in today's fast-paced business landscape.

And with local autonomy comes speed and agility: By distributing data ownership and fostering collaboration, Data Mesh enables teams to respond swiftly to changing business needs and data requirements. Domain teams have the flexibility to adapt their data products and infrastructure to meet evolving demands, avoiding the constraints of rigid centralized systems. This agility empowers organizations to seize emerging opportunities, make data-driven decisions in real-time, and stay ahead of the competition.

And perhaps the most interesting by-product of agility is establishing a culture of innovation and experimentation. With local autonomy, teams are encouraged to explore new ideas, test hypotheses, and iterate on their data products. This fosters a sense of ownership and accountability, spurring creativity and driving continuous improvement.

By embracing Data Mesh principles, organizations can unlock the potential of their data assets, enabling teams to discover valuable insights, develop innovative solutions, and drive business growth.

Today's Data Challenge

But what problems will Data Mesh and its promise of "agile data" address?

Consider data silos. Data silos hinder data accessibility and collaboration, making it difficult to gain a holistic view and leverage the full potential of the available data. They present a real, present, and formidable challenge that almost all data practitioners experience in modern enterprises.

Data silos, much like isolated islands in an immense ocean, are repositories of data that are confined within specific

departments or systems, disconnected from the broader organizational data landscape. This segregation results in a fragmented data ecosystem, where valuable insights remain untapped, and the collective intelligence of the enterprise is underutilized.

The existence of these silos often stems from historical organizational structures, disparate technology platforms, and departmental boundaries that have solidified over time. As a result, critical business decisions are frequently made based on incomplete or outdated information, leading to inefficiencies, missed opportunities, and a weakened competitive edge.

The ramifications of data silos extend beyond mere inefficiencies; they actively hinder collaboration and innovation within an organization. When data is trapped in silos, it becomes difficult for teams to access the information they need to collaborate effectively. This lack of accessibility and visibility leads to duplicated efforts, inconsistent data practices, and a general sense of organizational disjointedness.

And, in today's data-driven business environment, the inability to integrate data from different parts of the organization can impair a company's ability to respond to market changes, understand customer needs, and optimize operations. The

challenge is compounded in organizations with a global footprint, where the diversity of data sources, regulations, and business practices adds layers of complexity to the already intricate task of data integration and harmonization. Overcoming the challenge of data silos requires a strategic and concerted effort to foster a culture of data sharing and collaboration. This involves not just the adoption of new technologies but a fundamental shift in organizational mindset and practices.

A unified data strategy, underpinned by principles of open communication, shared goals, and cross-departmental collaboration, is essential. Such a strategy should encompass the establishment of common data standards, governance models, and a central data architecture that facilitates the seamless flow of information across different parts of the organization.

In this context, approaches like Data Mesh become highly relevant, offering a decentralized, yet cohesive framework for data management. Data Mesh advocates for domain-driven ownership of data, enabling individual teams to manage and share their data effectively while aligning with the overall organizational objectives. By embracing this paradigm, enterprises can gradually dismantle the barriers of data silos,

paving the way for a more integrated, agile, and data-centric organizational culture.

Now, let's consider data complexity. In the digital era, the complexity of data - and its management - has become a central challenge for modern enterprises. This complexity is not merely a byproduct of the volume of data but also its variety and velocity.

Data streams into organizations from an array of sources: social media feeds, IoT devices, customer interactions, enterprise systems, and more. This "digital exhaust" contributes to a rich, yet overwhelmingly intricate tapestry of information. This data landscape resembles a densely populated urban sprawl, with intricate networks and interdependencies that are challenging to map and manage. The task of understanding data lineage, dependencies, and relationships in such a diverse environment is akin to navigating a maze without a map, where each turn can reveal new paths or dead ends. This complexity not only strains traditional data management approaches but also demands new strategies and tools to make sense of this ever-expanding data universe.

The consequences of this complexity are far-reaching, impacting every facet of an organization. Decision-making

becomes encumbered when data is not easily accessible or interpretable. The challenge intensifies when dealing with unstructured data, which often holds valuable insights but is more difficult to analyze than structured data.

And as data volume and variety grow, ensuring data quality and integrity becomes an increasingly difficult task. Poor data quality can lead to incorrect or bad business decisions, misguided strategies, and ultimately, a detrimental impact on business outcomes. Making matters worse, the sheer complexity of data can obstruct compliance efforts, as understanding the nuances of data privacy regulations becomes more difficult when data is scattered and convoluted. For global organizations, this challenge is amplified by the need to navigate a patchwork of regional and international data laws.

Mastering this complexity requires a multifaceted approach, blending technology, strategy, and organizational culture. Advanced technologies such as machine learning and artificial intelligence offer powerful tools for analyzing complex data sets, uncovering patterns, and generating insights that would be impossible for humans to discern unaided. However, technology alone is not a panacea; it must be coupled with a robust data strategy that prioritizes data governance, quality, and integration. Organizations need to foster a data-literate

culture where employees across departments understand the importance of data and are equipped with the skills and tools to leverage it effectively.

But as importantly, a shift towards more agile, flexible data architectures, such as those advocated by Data Mesh, can also play a crucial role. By decentralizing data ownership and management, Data Mesh allows domain-specific teams to handle their data more effectively, reducing bottlenecks and enhancing responsiveness to change. This approach not only helps manage complexity but also empowers teams to extract maximum value from their data, turning a potential obstacle into a strategic asset.

Now, let's consider a challenge that bombards data professionals on a daily basis: data security.

Data security in today's digital landscape is an ever-shifting battlefield, with new threats emerging as rapidly as the technologies designed to combat them. For modern enterprises, safeguarding sensitive data against breaches, unauthorized access, and cyber attacks is not just a technical challenge but a critical business imperative.

The complexity of this task is magnified by the sheer volume and diversity of data that organizations handle. With data collected from myriad sources – customer databases, online transactions, IoT devices, and more – the risk of exposure and vulnerability increases exponentially.

And, the regulatory landscape adds another layer of complexity, with stringent requirements like GDPR (the European Union's General Data Production Regulation), HIPAA (Health Insurance Portability and Accountability Act), and others imposing strict guidelines and constraints on data handling, privacy, and protection. Navigating this intricate web of regulations demands not only robust security infrastructure but also a vigilant and proactive approach to data management and governance.

Data Mesh offers a paradigm shift in how data security is approached in large organizations. At its core, Data Mesh emphasizes decentralized data product ownership, ensuring clear accountability for each data set within an enterprise.

This model contrasts sharply with traditional centralized data management systems, where the responsibility for data security is often diffused among various departments, leading to potential lapses and slow response times. In a Data Mesh

framework, each domain team becomes the custodian of its data products, imbuing them with a deeper understanding and commitment to the security of their data.

This clear ownership not only fosters a more focused and vigilant approach to securing data but also enables swifter action in addressing urgent security issues. When security threats emerge, domain teams can respond rapidly, applying targeted measures to protect their data products without the delays often associated with centralized decision-making processes.

Addressing the complexities of data security in the modern enterprise requires more than just technological solutions; it demands an in-depth understanding of the data itself and the relevant security and privacy policies. Domain teams in a Data Mesh framework are uniquely positioned to meet this requirement.

By virtue of their close involvement with their data products, these teams possess detailed knowledge of the data's nature, use cases, and potential vulnerabilities. This intimate understanding enables them to implement security measures that are precisely tailored to the specific characteristics and risks associated with their data.

And the decentralized nature of Data Mesh facilitates a more adaptive and responsive security posture. As domain teams are deeply versed in the relevant legal and regulatory requirements, they can ensure compliance in a dynamic legal landscape, adapting quickly to new regulations and privacy standards. This approach not only strengthens the security of data assets but also enhances the overall resilience of the organization against the myriad threats that define the contemporary data security landscape.

But what about the characteristics of the data itself?

In the digital age, the velocity of data creation and consumption has become a defining challenge for organizations. Data is generated at an unprecedented rate from a plethora of sources – social media, mobile devices, IoT sensors, and countless enterprise applications. This rapid generation and consumption of data, akin to a high-speed train, necessitates a continuous and agile approach to data management.

Traditional data infrastructures often struggle to keep pace, leading to bottlenecks and delays in data processing and analysis. The challenge is not just in storing this vast amount of data but in processing and extracting value from it in real-time. Organizations need to adapt their infrastructure, tools, and

processes to not only manage this deluge of data but to leverage it effectively for timely decision-making and insights.

Data Mesh offers a compelling solution to the challenge of data velocity. By its very design, Data Mesh is oriented towards handling large volumes and high velocities of data efficiently. It does so by decentralizing data ownership and management. In a Data Mesh framework, data is no longer a centralized asset to be managed from a single point. Instead, it is distributed across multiple domain-specific teams, each equipped with the tools and autonomy to manage their slice of the data ecosystem.

This decentralized approach allows for distributed teams to process data independently thereby significantly reducing the time it takes to ingest, process, and analyze data. And, by empowering domain teams, Data Mesh ensures that data handling is more responsive and aligned with the specific needs and dynamics of each domain, enabling faster and more effective decision-making.

The agility and responsiveness that Data Mesh brings to data management are crucial in an era where the speed of data is continuously accelerating. As domain teams are closer to the data sources and more attuned to the specific requirements of

their domain, they can implement more effective and timely data strategies.

This not only includes the technical aspects of data handling but also encompasses a more nuanced understanding of the data's context and potential value. Data Mesh's emphasis on viewing data as a product means that each data product is designed with its use cases in mind, ensuring that it is not just processed efficiently but also utilized effectively.

By adopting Data Mesh, organizations can transform the challenge of data velocity into an opportunity, leveraging the rapid flow of data to drive innovation, enhance customer experiences, and make more informed, agile business decisions.

And last but not least, comes every data practitioner's favorite topic: data governance.

Data governance is an indispensable component in the architecture of modern enterprise data management, primarily due to the need to adhere to regulatory, privacy, and enterprise security policies. Effective governance ensures that data is managed and utilized in a way that meets these external and internal requirements. This includes compliance with legal

frameworks like GDPR and HIPAA, adherence to privacy norms, and alignment with organizational security protocols.

Given the penalties for non-compliance and the risks associated with data breaches, governance is not just a compliance issue but a critical business necessity. In this evolving landscape, data governance must be agile, responsive, and deeply integrated into the day-to-day handling of data.

Traditionally, data governance has often been managed through centralized models. While such models offer uniformity and central control, they frequently lead to slow and bureaucratic practices, creating bottlenecks that hinder the dynamic use of data. In centralized governance systems, decisions about data access, quality, and security are often made by a detached central authority, far removed from the context in which the data is used.

This distance can lead to inefficiencies and misalignments between governance policies and the actual needs and realities of different business units. The result is often a governance model that is seen more as a hindrance than an enabler, slowing down innovation and responsiveness to changing business and market demands.

But far too often, today data governance is viewed as a task that must be done, a command from on-high, rather than a task that drives inherent value. Data Mesh offers an alternative.

Data Mesh addresses data governance challenges by advocating for a federated governance model, which positions accountability for governance with the data owners who are most knowledgeable about the data. In this model, governance is decentralized, with each domain team responsible for the governance of its data products. This approach ensures that governance decisions are made by those who have the deepest understanding of the data's context, use, and risks. It leads to more relevant, efficient, and effective governance practices that are closely aligned with the specific needs of each domain.

To better understand the federated governance model of Data Mesh, consider an analogy with the American Standards Association (ASA, a US-based product standards organization, but almost every country or region has an equivalent organization). In this context, the ASA sets rules and policies and offers a certification process that allows vendors to ensure their products meet established standards. This certification process acts as a "brand" or "logo" of trust. Vendors can then publish their certification status, signaling to consumers that their products meet high standards.

In the Data Mesh governance model, the data governance team is akin to the ASA, setting overarching governance standards and policies. Data product owners, on the other hand, are like the product vendors. They ensure that their data products comply with the established governance standards and, once compliant, can be certified as meeting the enterprise's governance criteria.

This certification not only serves as a mark of trust and quality within the organization but also streamlines the process of governance by empowering those closest to the data. It ensures that governance is not a top-down, bureaucratic process but a collaborative, integrated practice that enhances the value and security of data across the enterprise.

Furthermore, data product owners, who are closest to the data and its use cases, are in a unique position to understand and manage the compliance requirements effectively. They can publish and update their certification statuses, making this information transparent and accessible within the Data Mesh ecosystem.

This method contrasts starkly with the conventional centralized governance models, where compliance is often managed by a central group that oversees and polices all data activities. While

this model has its strengths in maintaining control and uniformity, it can also lead to bottlenecks, delays, and a disconnect between the governance process and the real-world application of data.

In a federated model, the responsibility for compliance is distributed, fostering a culture of accountability and agility among data product owners. They can respond more swiftly to changes in regulations or business needs, updating their certification status and ensuring that their data products remain compliant. This not only streamlines the governance process but also embeds compliance into the fabric of the Data Mesh, making it an integral part of the data product lifecycle rather than an external, enforced process.

Turning Principles into Practice

By now, hopefully you will see that Data Mesh offers clear benefits. But realizing these benefits means turning the revolutionary Data Mesh principles into practice. So, that is what we think the core purpose of this book is. This book is driven by three foundational goals, each carefully crafted to guide professionals on their journey to mastering Data Mesh.

Our first goal is to demystify the transition from Data Mesh theory to practice. We don't just discuss the principles abstractly; we illustrate them through real-world examples, detailed case studies, and practical strategies that can be directly applied in your organizational context.

Second, we aim to accelerate your journey through the Data Mesh landscape. Understanding the intricacies of Data Mesh is one thing; applying them efficiently and effectively is another. This book offers a suite of techniques and best practices, distilled from leading industry experts and pioneering organizations, to fast-track your Data Mesh implementation. We delve into advanced topics such as automating governance, optimizing data product design, and leveraging cutting-edge technologies to amplify the benefits of Data Mesh in your enterprise.

Third, our intention is to chart a clear, actionable roadmap to Data Mesh success. This roadmap is more than a theoretical guide—it is a practical toolkit that addresses the common challenges and pitfalls encountered in implementing Data Mesh. From establishing a robust self-serve data infrastructure to nurturing a data-oriented culture, we provide a step-by-step guide to navigate the complexities of Data Mesh, ensuring a smooth and successful journey from inception to execution.

In embracing these principles and translating them into actionable practices, we envision a future where organizations can fully harness the transformative power of Data Mesh. We believe that the adoption of Data Mesh principles can propel data initiatives to unprecedented heights, enabling businesses to become more agile, data-driven, and competitive.

Our aspiration in writing this book is rooted in a humble yet bold vision: two decades from now, we hope to look back and see Data Mesh as a pivotal force in bringing agile methodologies to the realm of data management. Our contribution, though a modest part of this larger movement, aims to empower organizations to derive better, faster, and more cost-effective insights and business value from their data. Through the pages of this book, we seek to inspire a new generation of data professionals, equipping them with the knowledge and tools to revolutionize data management practices and drive their organizations towards a future where data is not just an asset, but a catalyst for innovation and growth.

In today's data-driven landscape, organizations face a myriad of challenges when it comes to managing and harnessing the power of data. The sheer volume and variety of data sources can be overwhelming, resembling an overflowing river that organizations struggle to navigate. Making sense of this deluge

of data, ensuring its quality, and extracting valuable insights pose significant hurdles.

Zhamak Dehghani's Data Mesh principles offer a revolutionary vision for data management. It advocates for decentralized ownership, self-serve data platforms, federated computational governance, and cross-functional collaboration. By applying agile principles to data, Data Mesh promotes local autonomy, speed, and agility.

And by translating these principles into practice, organizations can overcome these challenges and unlock the benefits of Data Mesh, providing improved data accessibility, quality, and responsiveness to changing data demands.

The remainder of this book aims to provide practical guidance on implementing Data Mesh, establishing self-serve data infrastructure, fostering a data product mindset, implementing federated data governance, creating decentralized ownership, promoting cross-functional collaboration, and facilitating knowledge sharing within organizations. The remainder of this book will touch upon several topics:

Establishing a self-serve data infrastructure

We will define Data Products (Chapter 2), and how they are members in the Data Mesh ecosystem. We will introduce our case study (Chapter 3) - applying Data Mesh to make climate data easy to find, consume, share, and trust - that will be used throughout the book to demonstrate how to implement Data Mesh practices. And, of course, we will have a perspective on Data Mesh architecture (Chapter 4).

Embracing a data product mindset

We will describe how Data Contracts (Chapter 5) let all members of the Data Mesh ecosystem find each other and interact. We will explain how to encourage domain teams to think of data as a product and define clear data product boundaries, establish APIs, documentation, and support mechanisms required for your first Data Product (Chapter 6, 7, and 8). And we will describe a “test and learn” mindset that encourages teams to iterate and improve their data products based on feedback and evolving business needs, and promote a culture of continuous improvement and innovation within each Data Product team.

Driving value from generative-AI in Data Products

Generative-AI - OpenAI and ChatGPT and their open source counterparts - promise to shake the foundations of the modern enterprise. Data Mesh, obviously, is no different. In fact, we see material and widespread uses for generative-AI (Chapter 9) that we will explain.

Creating a domain-oriented decentralized ownership

We will describe how to run and operate your Data Mesh (Chapter 10) and describe the "team topology" required to deliver your Data Mesh (Chapter 11) and show how to establish autonomous domain teams responsible for specific areas of the business and will describe the role of the Data Product owner that has ownership of their domain's data, including its quality, governance, and accessibility.

Promoting cross-functional collaboration

We will define and then describe the intricacies of an operating model for Data Mesh (Chapter 12). and then we will discuss the incentives and organizational structure that allows a Data Mesh to gracefully evolve and grow. We will also explain the shift from a centralized data governance model to a federated approach, offering an approach to automate Data Product governance to create

a “self-serve federated governance mechanism”. (Chapter NEW). And this will enable domain teams to define and enforce data governance policies within their domains while aligning them with the organization’s overall governance framework and provide guidance and frameworks to ensure consistent standards and compliance across domains while allowing flexibility for domain-specific requirements.

Creating your Data Mesh Roadmap

We will provide a tried and tested “roadmap” (Chapter 13) that starts with a strategy, and then shows how to implement the core Data Product and Data Mesh foundational elements as well as establishing Data Product teams and the broader Data Mesh operating model. We will also show how to establish channels for collaboration and knowledge sharing among domain teams through communities of practice, regular cross-functional meetings, or data councils. We will show how to socialize Data Mesh within your organization to encourage teams to share best practices, lessons learned, and data assets to leverage the collective knowledge and expertise across the organization.

By putting these principles into practice, organizations can overcome data management challenges and realize the benefits of Data Mesh. They can achieve local autonomy they crave and need, giving Data Product teams ownership and control over their data, allowing them to operate at a faster pace, leveraging self-serve infrastructure and enabling rapid iteration and experimentation. Finally, they can embrace agility by fostering collaboration, adopting a data product mindset, and implementing federated data governance. Through these practical steps, organizations can transform their data management approach and unlock the full potential of their data assets.

Enjoy!

Chapter 2. Applying Data Mesh Principles

A NOTE FOR EARLY RELEASE READERS

With Early Release ebooks, you get books in their earliest form—the author's raw and unedited content as they write—so you can take advantage of these technologies long before the official release of these titles.

This will be the 2nd chapter of the final book. Please note that the GitHub repo will be made active later on.

If you have comments about how we might improve the content and/or examples in this book, or if you notice missing material within this chapter, please reach out to the editor at sevans@gmail.com.

For those that have read Zhamak Dehghani's visionary book, this will be a review of sorts. But for those that are new to Data Mesh, our hope is that this is a valuable - but brief - introduction to Data Mesh, its core principles, and its core components.

What is a Data Mesh?

At its simplest, a Data Mesh is just an ecosystem of interacting data products as shown in Figure 2-1. But like any ecosystem, there are many moving parts, each operating somewhat independently, that are connected through common standards and a communications fabric. And data products in a data mesh, ideally, have a common technical implementation with a consistent set of interfaces.

So, Data Mesh is, then, at its foundation, a conceptual framework in the realm of data architecture, which emphasizes decentralized data ownership and architecture. It recognizes that in large organizations, data is vast and varied, where each business domain has autonomy over its own data. By decentralizing control, it empowers individual domains to manage and make decisions about their data while maintaining a cohesive overall structure. And presumably, with this autonomy comes better, more localized, and faster decisions, which in-turn, leads to speed and agility.

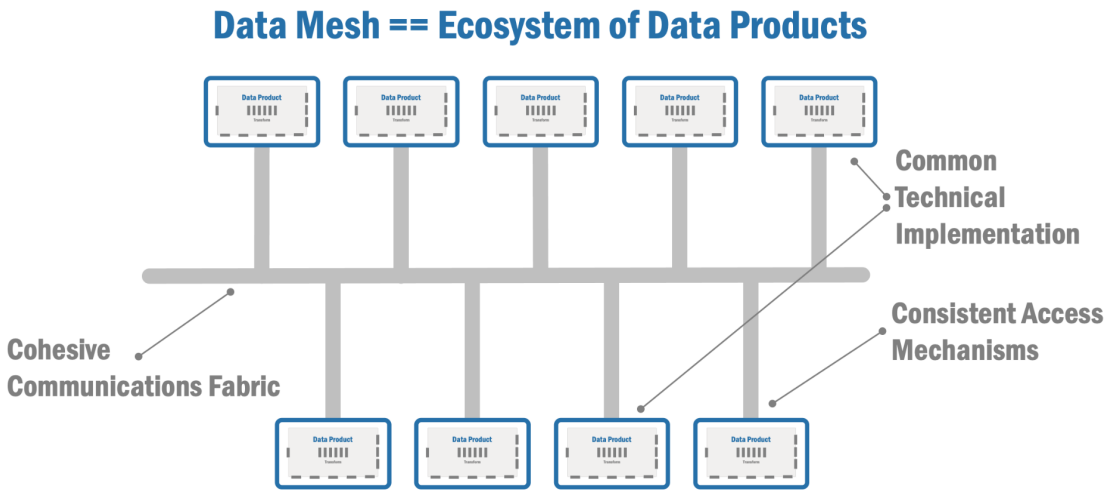


Figure 2-1. :Data Mesh - An Ecosystem of Interacting Data Products

In the context of a Data Mesh, a data product is a package of data that is self-contained, self-descriptive, and oriented towards a specific business purpose or function. But, it is not just a mere collection of data, rather it is a coherent unit that provides value to its consumers, akin to a product in any other industry.

Data products are sophisticated packages of data, uniquely crafted to address specific business objectives within an organization. These are not mere collections of data; rather, they are comprehensive units that encapsulate the data itself along with the essential tools, documentation, and metadata. This composition ensures that the data is not only present but

also understandable and usable. Each data product is purpose-oriented, tailored to serve a particular business need or solve a specific problem, making them much more than just repositories of information.

The structure of a data product is self-contained, meaning that it includes everything necessary for its effective utilization. It adheres to strict standards of quality and governance, thereby ensuring reliability, security, and compliance with relevant regulations. This comprehensive approach makes data products a trusted and dependable resource within the organization. They are designed with user accessibility in mind, offering interfaces and documentation that are easily navigable by a wide range of users, from data experts to those with minimal technical expertise.

Furthermore, the life cycle of each data product is meticulously managed. Every product has an assigned owner, responsible for its maintenance, updates, and overall management. This stewardship ensures that the data product remains relevant and continues to deliver value over time. The continuous oversight and improvement of these data products underpin their evolving nature, ensuring they stay aligned with the dynamic needs and objectives of the organization. This life cycle management is a critical aspect of data products,

distinguishing them as not just static data sets, but as evolving assets within the Data Mesh ecosystem.

Obviously, we will have much more to say about the Data Mesh ecosystem in the architecture chapter (Chapter 4). Nevertheless, let's continue.

Data Mesh Principles

A set of guiding principles stands at the core of Data Mesh, with each playing a crucial role in the framework's efficacy and sustainability. The first of these principles is the establishment of a clear boundary for each data product. This boundary demarcation is essential in defining what each data product represents, its scope, and its limitations. It's an exercise similar to mapping out a city's boroughs, where each area is distinctly outlined and understood for its unique characteristics and contributions. The principle of a clear boundary in a Data Mesh ensures that every data product is a well-defined entity within the larger ecosystem. This clarity prevents overlap and confusion, establishing a clear understanding of the data product's purpose and scope. It aids in managing expectations and directs efforts and resources appropriately, ensuring that each data product can effectively fulfill its intended role.

Another fundamental principle in the Data Mesh framework is the concept of an empowered owner for each data product. This aspect of the framework borrows from the idea of having a dedicated manager for each city block, someone who is deeply invested and responsible for its well-being. In a similar vein, each data product within a Data Mesh has an owner who bears the responsibility for its performance, quality, and compliance with governance standards.

The role of an empowered owner is multifaceted. They are tasked with ensuring that the data product aligns with both specific business requirements and the overarching governance framework. This alignment is crucial for maintaining the integrity and usefulness of the data product, ensuring it remains a valuable asset within the organization's data landscape.

A third principle central to the Data Mesh concept is the provision of self-serve capability. This feature is pivotal in democratizing data within an organization. It allows users from various departments and skill levels to access, manipulate, and analyze data independently, without relying on specialized technical support. This approach to data accessibility is comparable to empowering city residents to utilize public

spaces and services on their own terms, fostering a sense of ownership and engagement.

Self-serve capability in a Data Mesh not only empowers users but also fosters a culture of innovation and agility. It enables individuals to leverage data for their specific needs, encouraging experimentation and personalized analysis. This capability reduces bottlenecks typically associated with centralized data systems, where requests for data access and analysis can slow down decision-making processes.

The final principle guiding the Data Mesh framework is federated computational governance. This principle underscores a distributed approach to governance, establishing a cohesive set of rules and standards that all data products within the Mesh adhere to. This approach is analogous to citywide regulations that ensure harmony and order across diverse boroughs and neighborhoods.

Federated computational governance is essential for maintaining consistency and compatibility across the Data Mesh. It ensures that despite the decentralized nature of data ownership, there is a unified framework governing how data is managed, used, and shared. This unified approach is crucial in

preventing data silos, ensuring data interoperability, and maintaining the overall integrity of the data ecosystem.

Implementing federated computational governance requires a delicate balance. It involves creating governance structures that are robust enough to ensure consistency and compliance, yet flexible enough to accommodate the unique needs and contexts of different data products. This balance is key to fostering an environment where innovation can thrive without compromising the standards and protocols essential for a cohesive data ecosystem.

Defining a “Good” Data Product

As stated earlier, a Data Mesh is an ecosystem of Data Products. So, practically speaking, a data product is the foundational building block and, in fact, the smallest indivisible unit, a “data quantum” of sorts, for any Data Mesh. So, clearly Data Products are crucial, and we better ensure that they are “good”. But what does “good” look like, or more specifically, what is the definition of a “good” data product? Clearly there are many attributes that span technical, business, and ease of use, and many other characteristics that constitute a “good” data product, as shown in Figure 2-2.

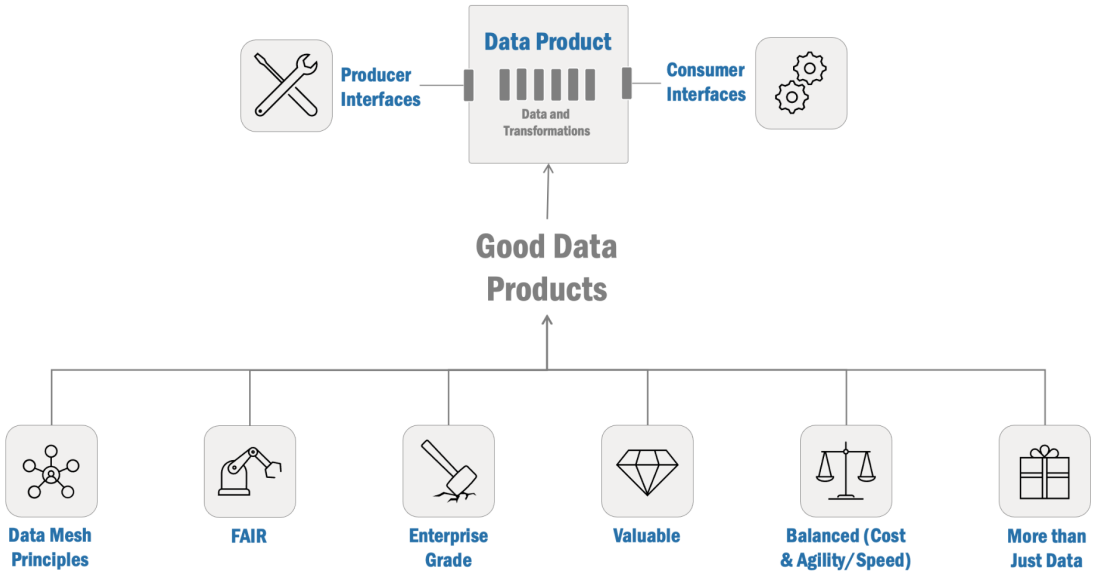


Figure 2-2. : Good Data Products

Defining a Principled Data Product

So, where to start? Let's start with a simple and perhaps obvious statement: Good data products adhere to data mesh principles. Let's look at these principles and apply them to data products. First, good data products align to decentralized domain ownership - they should align to a domain (large or small) with a clear boundary, and should have an empowered owner. Second, good data products are treated as their name implies - as a product and not a project. Good data products, like other products, have a life cycle, have clear consumers, and a clear value proposition. Good data products are self-serve,

meaning that users can get what they need from a data product without undue participation from third parties. Good data products also have a federated governance mechanism, that is to say, that provides for local autonomy and decision making - at the data product level by the data product owner and their team. It also means that while being federated, data product owners/teams are responsible and accountable to ensuring their data product adheres to enterprise guidelines and standards as needed.

Defining a FAIR Data Product

Good data products should also adhere to FAIR principles: Data should be:

- Findable
- Accessible
- Interoperable
- Reusable

According to FAIR, “the principles emphasize machine-actionability (i.e., the capacity of computational systems to find, access, interoperate, and reuse data with none or minimal human intervention) because humans increasingly rely on

computational support to deal with data as a result of the increase in volume, complexity, and creation speed of data."

So, let's elaborate on these principles and apply them to data products. Findability is the first of the FAIR principles. For a data product to be valuable, it must be easily discoverable within the organization's broader data landscape. This involves implementing a system for cataloging and effective metadata management. For instance, a data product containing sales figures should have detailed metadata that includes information on the time period covered, the geographical scope, and the type of sales data included. This enables users to quickly find and identify the data they need without wading through irrelevant information.

Accessibility is another key principle. It's not enough for data to be findable; In fact, once found, a data product must also be easily accessible. Accessibility includes providing comprehensive documentation that explains how to use the data, as well as ensuring that the data can be easily integrated into various applications and workflows. A good data product should be as straightforward to use as a well-designed software application, with clear instructions and support.

Interoperability is a critical aspect of FAIR principles. It refers to the ability of data products to work together and integrate effectively. In practical terms, this means that data products should be created using standard data formats and protocols. For example, if one data product uses XML format and another uses JSON, there should be tools or services in place to allow these different formats to be used together seamlessly. This interoperability is essential for combining and leveraging data from various sources.

The fourth principle, Reuse, focuses on the ability to apply data in multiple contexts. This principle is particularly important in maximizing the value of data. By designing data products to be modular and reusable, they can be used across different projects and applications. For instance, a data product containing customer demographic information can be used by marketing teams for campaign planning, by sales teams for sales strategy development, and by product development teams for market analysis.

In addition to these technical aspects, adhering to FAIR principles also involves fostering a culture of collaboration and data sharing. This cultural shift is crucial in breaking down silos and encouraging the reuse of data products. It means

promoting an organizational mindset where data is seen as a shared resource that can be leveraged for multiple purposes.

In conclusion, “good” data products in a Data Mesh are those that are FAIR: findable, accessible, interoperable, and reusable. These principles ensure that data is not just stored but is actively managed and used in a way that adds value to the organization. By adhering to FAIR principles, data products become more than just repositories of information; they transform into dynamic assets that drive innovation and decision-making across the enterprise.

Defining an Enterprise Grade Data Product

So, FAIR principles provide one lens to understand a “good” data product. But what makes a data product “good” in an enterprise? Or more specifically, what is the definition of a “Good” Enterprise Grade data product? Well, I suppose in the realm of enterprise-grade data products, several key attributes come together to define their quality and effectiveness. These attributes, encompassing security, reliability, observability, operability, deployability, and comprehensive documentation,

form a cohesive structure that ensures the data product's value within an organization.

But practically, the strength of an enterprise-grade data product lies in the seamless integration of its key attributes. Security, reliability, observability, operability, deployability, and comprehensive documentation are not isolated aspects; they are interconnected, each playing a vital role in the product's overall functionality and value. A product that excels in these areas is not just a repository of data but a dynamic asset that drives business efficiency, innovation, and decision-making. So, understanding how these attributes interplay and support each other is crucial in creating a data product that meets the stringent demands of enterprise environments.

Security probably stands at the forefront of these enterprise-grade attributes. An enterprise-grade data product must be fortified against unauthorized access and breaches, ensuring the confidentiality and integrity of the data it holds. This security is not only about safeguarding information but also about maintaining user trust and adhering to regulatory standards such as GDPR or HIPAA. Implementing robust encryption, access controls, and regular security audits are integral to this process, creating a fortified barrier against potential cyber threats.

Yet, security alone is not sufficient. The reliability of the data product is equally important. Users need to trust that the data product will provide accurate and consistent information at all times. Ensuring reliability involves implementing validation checks, error detection algorithms, and maintaining high data availability. This is where the concept of reliability intersects with security; a secure data product is inherently more reliable as it protects against data tampering and loss.

Observability extends the concept of reliability. It's about having the ability to monitor the health and performance of the data product. By using tools to track various metrics like response times and error rates, organizations can proactively manage the data product's health. This proactive management plays a crucial role in maintaining the product's reliability, as it allows for the early identification and resolution of potential issues before they escalate.

Closely linked to observability is the aspect of operability. A data product with high operability is easier to manage and operate. This involves capabilities that streamline the data product's lifecycle management, including deployment, scaling, updating, and troubleshooting. High operability supports the product's reliability by ensuring that it remains functional and

effective throughout its lifecycle, adapting to changing requirements with minimal disruption.

Deployability is another critical attribute, especially in dynamic business environments. A highly deployable data product can be easily implemented and integrated into various business processes and technological environments. This flexibility is crucial for keeping pace with the evolving needs of a business, whether it's scaling to accommodate growth or integrating with new systems and applications.

Underpinning all these attributes is the role of comprehensive documentation. Documentation serves as the backbone of a data product, providing clarity on its use, management, and integration. It includes everything from user guides and API documentation to operational procedures and architectural diagrams. Good documentation not only aids in the effective utilization of the data product but also ensures compliance with regulatory standards, facilitating audits and compliance checks.

The interplay between these attributes creates a holistic enterprise-grade data product. For instance, robust documentation enhances security by outlining precise data handling procedures, while observability informs reliability strategies by identifying predictive maintenance needs.

Similarly, the ease of operability is often facilitated by well-structured documentation, which provides clear guidelines for managing and updating the data product.

User experience is another crucial aspect that ties these attributes together. An enterprise-grade data product should not only be robust and reliable but also intuitive and user-friendly. This involves considering diverse user needs and incorporating thoughtful UI/UX design, ensuring that the product is accessible to a wide range of users with varying technical expertise.

Defining a Valuable Data Product

Beauty is in the eye of the beholder, as they say. Nevertheless, there are a few objective characteristics of a data product that we can use to clearly and unambiguously attribute value. First, a valuable data product is fundamentally defined by its relevance and utility. The primary purpose of such a product is to address specific business needs or questions, making it a crucial tool for informed decision-making and insight generation. The value is directly tied to its practical application in solving real-world business problems or enhancing operational efficiency. Therefore, a data product's utility is

gauged by its ability to facilitate actions, decisions, or provide insights that are directly applicable to the users' needs.

Quality and reliability are indispensable attributes of a valuable data product. This encompasses not only the accuracy, consistency, and completeness of the data but also its timeliness and relevance to current business scenarios. Furthermore, reliability extends to the technical aspects of the data product, including its performance capabilities like processing speed and availability. Ensuring high quality and reliability is crucial as these factors directly impact the trustworthiness and dependability of the data product in operational and decision-making processes.

Usability is also a critical attribute. Usability is a key determinant of a data product's value - if it is complex or unintuitive, its potential utility diminishes irrespective of the underlying data quality. Therefore, the design and interface of a data product should facilitate ease of use to ensure that it can be effectively employed by its target users. Somewhat related to this is interoperability - in other words it is also usable from an operations perspective. A valuable data product should not only function in isolation but also integrate seamlessly with other data products. This interoperability is vital for comprehensive analytics and insight generation, as it allows for the

combination and analysis of data across various domains. Additionally, compliance with regulatory requirements and security standards is non-negotiable. Ensuring data privacy, adherence to regulations like GDPR or HIPAA, and maintaining robust security protocols are fundamental to the integrity and value of a data product.

Lastly, scalability and maintainability are key aspects of a valuable data product. It should be capable of handling increasing volumes of data or user demands without necessitating significant redesign or rework. Alongside scalability, maintainability - the ease with which a data product can be updated, modified, or repaired - is critical for its long-term utility. This also includes the product's ability to evolve based on user feedback and changing business needs, ensuring that it remains relevant and valuable over time. Aligning with the organization's strategic objectives and contributing to business goals, whether through cost reduction, revenue generation, or risk management, solidifies a data product's value within the organization's ecosystem.

Defining a Balanced Data Product

A key attribute of a valuable data product is achieving this balance between cost and efficiency, and speed and agility. Now, there is always a delicate balance to be struck between cost control and the pursuit of speed and agility. Traditionally, IT organizations - especially those that are highly centralized - have leaned heavily towards optimizing for cost control. This focus, while financially prudent, often comes into conflict with the business's growing need for speed and agility - attributes that are increasingly crucial in today's fast-paced market environment.

However, a shift in perspective reveals an interesting dynamic. In fact, experience has shown prioritizing speed and agility doesn't necessarily compromise cost-effectiveness and by focusing on these aspects, businesses can achieve more efficient project completion which can lead to cost savings in the long run. This efficiency is born out of the ability to adapt quickly to market changes, customer needs, and new technological advancements, thereby reducing the time and resources spent on lengthy project cycles.

This does, however, suggest or at least imply a particular (perhaps agile) continuous test and learn approach to developing a valuable data product that best balances cost/efficiency and speed/agility. This approach involves

incremental development, where data products, or their constituent capabilities, are broken down into smaller, manageable delivery units. This allows for rapid iteration and adaptation based on feedback and changing requirements. The use of prototypes and minimum viable products (MVPs) is central to this approach, enabling teams to test ideas and concepts without committing extensive resources to full-scale development. Obviously, organizations that are new to data mesh should weigh this consideration and approach quite heavily.

The incremental approach has several benefits. Firstly, it allows for quicker response times to market demands and customer feedback, as changes can be implemented and tested in shorter cycles. Secondly, it reduces the risk associated with larger data products since adjustments can be made along the way, avoiding the costly pitfalls of fully committing to a single, rigid delivery plan.

A “Good” Data Product is More than Just Data

Data products in a Data Mesh are often perceived as revolving exclusively around their data. However, this view is somewhat

limited. While data is indeed the foundational element, a data product encompasses much more, transforming it into a resource that is significantly more valuable than just the sum of its data parts. This expanded view is crucial in understanding the true potential and functionality of data products in a modern data environment.

An essential aspect of data products is the inclusion of a broader set of “artifacts.” In this context, artifacts are any objects, entities, or items that the data product owner decides to make available to its users or a broader audience. These artifacts extend the utility and applicability of the data product beyond its core data, adding layers of functionality and insight, as you see in Figure 2-3.

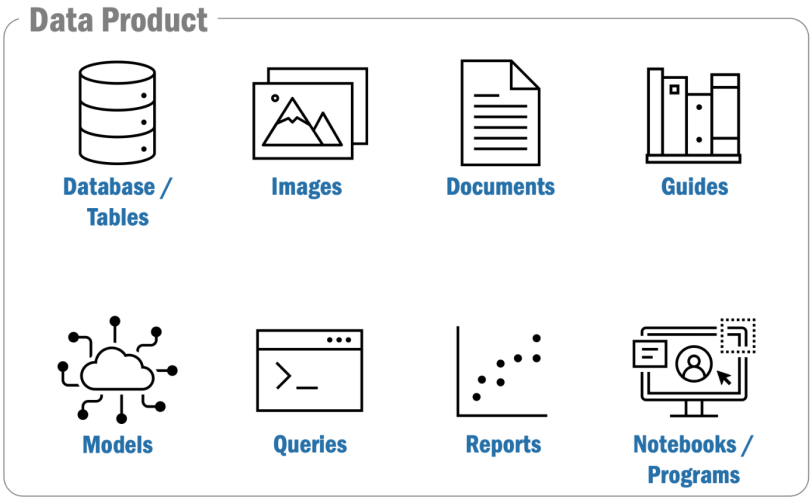


Figure 2-3. :Data Product Artifacts

Programs are a prime example of valuable artifacts. They can demonstrate how the data within the product can be used effectively. These programs might include ’starter kits’ for users of the data product, providing them with a foundational understanding of how to interact with and extract value from the data. In some cases, these programs might showcase key insights or analyses derived from the data, offering users a head start in their exploration.

Another significant type of artifact is AI/ML models. In many modern data products, the data is used to train machine learning or artificial intelligence models. These models, when included as artifacts, can offer unique insights that are a direct

result of the data analysis. They serve as powerful tools for users who are looking to leverage advanced analytics within their own operations.

Queries are also an essential artifact in data products. These can include pre-written SQL queries or other access methods that provide users with ready-to-use insights. These queries are particularly valuable for users who may not have deep technical expertise but need to derive meaningful information from the data product.

Streams represent a dynamic aspect of data products. Users can subscribe to specific topics within the data product and receive notifications when there are changes or updates to the data. This feature is particularly useful for keeping users engaged and informed about the latest developments or insights derived from the data.

The inclusion of these diverse artifacts transforms the data product from a static repository of information into a dynamic toolkit for analysis and insight. The value of the data product is thus not just in the raw data it contains but also in how it enables users to interact with and make use of that data in varied and meaningful ways.

Moreover, the choice of artifacts to include in a data product is an important strategic decision. It reflects the data product owner's understanding of the needs and preferences of their target audience. By carefully selecting and curating these artifacts, the owner can significantly enhance the usability and appeal of the data product.

The process of integrating these artifacts into the data product also demands thoughtful consideration. It involves ensuring compatibility among different elements and creating an intuitive user experience. This requires a deep understanding of both the technical aspects of the artifacts and the user journey within the data product.

In addition, the management of these artifacts over time is crucial. As the needs of users evolve and new technologies emerge, the artifacts within a data product may need to be updated, replaced, or expanded. This ongoing maintenance ensures that the data product remains relevant, useful, and competitive.

Furthermore, the documentation of these artifacts is equally important. Clear and comprehensive documentation aids users in understanding how to leverage each artifact effectively. This

documentation should be easily accessible and understandable, catering to users with varying levels of expertise.

In summary, a data product in a Data Mesh is a complex and multifaceted entity that extends far beyond the data it contains. By including a variety of artifacts such as programs, AI/ML models, bundles, queries, and streams, a data product becomes a versatile and dynamic resource. These artifacts add layers of functionality and insight, making the data product not just a source of data but a comprehensive toolkit for analysis, exploration, and innovation.

Overall, the inclusion of artifacts in data products represents a significant shift in how data is presented and utilized in modern data environments. It showcases the evolution of data management from static storage to dynamic, interactive platforms that empower users to derive greater value and insights from their data.

Defining a Valuable Data Product

So, we now have talked about the technical aspects of a “good” data product - that it adheres to Data Mesh and FAIR principles and has the attributes to be considered enterprise grade. But there are clearly other considerations, and perhaps first and

foremost of these are: does a data product deliver value, and how does it deliver value? Or more specifically, how do we know if we have a valuable data product?

Defining what constitutes a valuable data product in a Data Mesh environment requires a deep understanding of several key aspects. First and foremost, a valuable data product must address and solve a clearly defined and recognized problem. This problem-solving attribute is the cornerstone of its value, as it ensures that the data product is not just a collection of data but a tool that addresses specific business needs or challenges. In fact, the ability of a data product to provide actionable solutions is what should set it apart. This means that the product doesn't just present data; it offers insights and answers that can be directly applied to resolve the identified problem.

The process of problem-solving with a data product begins with a thorough understanding of the challenge at hand. This involves gathering insights from various stakeholders and analyzing existing data to get a comprehensive view of the problem. A deep dive into the problem helps in crafting a data product that is precisely tailored to address the specific nuances of the issue. Once the problem is clearly understood, the next step is to devise a practical and feasible solution. This is where the data product comes into play as a critical component of the

solution. The design of the data product should directly contribute to resolving the identified problem, leveraging data analytics, machine learning, or other relevant technologies.

The target state of the data product should be ambitious yet achievable. It needs to strike a balance between aspirational goals and practical realities. The target state should challenge the status quo but remain grounded in what is realistically attainable given the current technological capabilities and organizational context. So, clearly a valuable data product must also have a well-defined target state or end goal. This target state should reflect a clear vision of what the data product aims to achieve or contribute to within the organization. Establishing this target state ensures that the development of the data product remains focused and aligned with the intended objectives.

Clearly linked to the target state is the need for a roadmap - a way to get to the target state. It's a strategic plan that details the progression from the current state of the data product to its desired future state, including the technologies, resources, and timelines involved. This is clearly a big topic and much more detail will be available in the "Establishing a practical Data Mesh roadmap" chapter (Chapter 13). Developing the roadmap for the data product is a collaborative process. It involves input

from various teams, including data scientists, IT professionals, and business stakeholders. The roadmap should be flexible enough to accommodate changes and agile enough to respond to new findings or shifts in business priorities.

Now, recognizing the long-term nature of a data product, a strong and senior level of engagement from senior executives is crucial. Which is where our sponsor comes in. The sponsor is typically a high-level executive or decision-maker within the organization who champions the data product. Their support is crucial for aligning the data product with the organization's broader goals and strategies. The role of the sponsor extends beyond mere endorsement. They are instrumental in navigating organizational hurdles and advocating for the data product across various departments. Their influence can be pivotal in securing buy-in from different stakeholders within the organization, ensuring that the data product is integrated and utilized effectively.

But having a sponsor with the right level of influence is critical in ensuring that the data product doesn't become sidelined or lost among other organizational priorities. The sponsor's role involves not just securing funding but also ensuring ongoing support for the data product throughout its development and deployment. So, presumably with a sponsor comes a

sustainable mechanism for funding for the creation and operation of a valuable data product.

Funding ensures that the data product has the necessary resources for development, deployment, and ongoing maintenance. It's about having a financial plan that supports the entire lifecycle of the data product. The funding mechanism should align with the overall value proposition of the data product. It's important that the investment in the data product is seen in the context of the returns it promises, whether in terms of efficiency gains, revenue generation, or other strategic benefits.

The funding mechanism for the data product should be viewed as an investment in the organization's future capabilities. This perspective helps in justifying the expenditure and aligning it with the long-term strategic objectives of the organization. The funding should be structured in a way that allows for scalability and evolution of the data product as needs and technologies change.

A "Good" Data Product has an Empowered Data Product

An empowered Data Product Owner is pivotal to the success and effectiveness of a data product. In this sense, it is not a specific attribute of a valuable data product, but it is still a necessary condition for delivering a valuable data product. In fact, it is the data product owner that governs the determination of what is considered valuable. They determine the balance between cost/efficiency and speed/agility. Their local autonomy is fundamental to their decision making power to influence and guide a data product from its genesis to production. So, please indulge me a bit to, perhaps, state the obvious: you cannot have a valuable data product without an empower data product owner.

Let's dig a bit deeper. The Data Product Owner holds a position of significant responsibility and authority, overseeing the overall health, performance, and strategic alignment of the data product with the business's needs. Their role is multifaceted, encompassing various aspects of data product management, from conceptualization to implementation and ongoing maintenance. So, perhaps it goes without saying, but without an empowered data product owner, you do not have a "good" data product.

The Data Product Owner's responsibilities are comprehensive. They are charged with ensuring that the data product not only

functions correctly but also delivers value in alignment with business objectives. This involves a deep understanding of both the technical aspects of the data product and the business context in which it operates. Their responsibilities extend to overseeing the development process, managing the product lifecycle, and ensuring that the end product effectively meets the intended goals.

Accountability is a crucial aspect of the Data Product Owner's role. They are answerable for the outcomes produced by the data product. This means ensuring that the product meets all quality and compliance standards, and that it delivers the expected results. Their accountability extends to all stakeholders, including the technical team, business users, and senior management, requiring them to maintain transparency and open communication about the product's progress and performance.

Now, one of the critical powers vested in a Data Product Owner is decision rights. They have the authority to make key decisions regarding the development, deployment, and evolution of the data product. This includes decisions about features, functionalities, and the overall direction of the product. Their decision-making authority is essential for

maintaining the product's relevance and effectiveness in a rapidly changing business environment.

And with these decision rights, an empowered Data Product Owner also has a high degree of autonomy. This autonomy allows them to operate independently within the defined boundaries of the data product, making decisions and implementing strategies that foster innovation and agility. The autonomy granted to them is not unfettered but is balanced with the need for alignment with broader organizational goals and strategies.

But let's make this a bit more concrete. Here is a common scenario that illustrates the need for clear decision rights - it involves the selection of technology tools and platforms for the data product. More specifically, it is quite frequent for an enterprise to have a preferred set of tools and platforms that it mandates across its operations. However, the Data Product Owner might identify alternative tools that they believe would be more effective for their specific data product.

In such cases, if the principles of Data Mesh are adhered to, the decision rests with the Data Product Owner. They have the authority to choose the tools and technologies that best suit the needs of their data product. This autonomy is crucial for

ensuring that the data product is built with the most suitable and effective technologies.

However, this decision-making autonomy doesn't mean isolation from the rest of the enterprise. The enterprise, on its part, should focus on making its recommended tools effective, efficient, and user-friendly. The goal should be to create an environment where Data Product Owners see the value in using enterprise-recommended tools, not because they are mandated, but because they genuinely meet their needs.

So, once again, you cannot have a valuable data product without an empower data product owner.

Conclusion

So, at this point we understand what a "good" data product is: it follows data mesh principles and is aligned to FAIR principles. It is enterprise grade. It delivers real tangible value. It balances both cost as well as agility and speed concerns. It is much more than just data. And, it has an empowered owner that can deliver on the data product's promise,

So, with all of this going for it, the next obvious question is "how do I build a 'good' data product that has all of these

attributes?”. Well, that is a big question, and the next two chapters will kickstart the process: We will first try to introduce a scenario that is used throughout the book to show how to put these principles and characteristics into practice. And then we will do a deep dive on the architecture components of both a data mesh and its constituent data products.

Chapter 3. Our Case Study - Climate Quantum Inc.

A NOTE FOR EARLY RELEASE READERS

With Early Release ebooks, you get books in their earliest form—the author's raw and unedited content as they write—so you can take advantage of these technologies long before the official release of these titles.

This will be the 3rd chapter of the final book. Please note that the GitHub repo will be made active later on.

If you have comments about how we might improve the content and/or examples in this book, or if you notice missing material within this chapter, please reach out to the editor at sevans@gmail.com.

In this chapter we will introduce our case study - Climate Quantum Inc. - where we will apply data mesh capabilities to an important and pressing need: Climate Change.

But first, some background.

Climate change, an undeniable and pressing reality, permeates every aspect of our global society. As businesses grapple with its vast implications, they find themselves confronting a new frontier: the labyrinthine world of climate data. While the data holds invaluable insights, navigating its intricacies poses significant challenges.

First, climate data is an ever-evolving landscape: Climate data isn't static; it constantly morphs, influenced by countless variables from anthropogenic activities to natural phenomena. Pinpointing the relevant data amidst this fluidity is no small feat.

Second, climate data volumes are huge, and incredibly diverse: The sheer volume of climate data is staggering. With thousands of data sources each governed by their licensing terms, some publicly available and others proprietary, the challenge lies not just in data acquisition but in harmonizing and interpreting it.

Moverover, the regulatory environment is changing fast: The tightening regulatory environment, with its expanding scope, mandates businesses to report with precision. Understanding and addressing the demands of Scope 1, 2, and especially the nuanced Scope 3, necessitates robust climate data frameworks.

And if that wasn’t enough, the axiom upon which foundational decisions are made - that past behavior is indicative of future behavior - is proving to be false. Historically, the past has been our compass, guiding predictions and decisions. However, the accelerating pace of climate change makes the past an unreliable predictor, thrusting businesses into largely uncharted waters.

Lastly, stakeholder and consumer expectations are also changing: As stakeholders champion transparency and consumers champion sustainability, businesses find themselves under increased pressure to validate their environmental credentials.

To further elaborate, the inherent complexity of climate data is compounded by its multidisciplinary nature. It encompasses a wide range of fields, from meteorology to oceanography, from glaciology to environmental science. Each of these disciplines generates massive amounts of data, often in different formats and scales, making integration and analysis a daunting task.

Additionally, the accuracy and reliability of climate data are paramount. Decisions made on the basis of this data can have far-reaching consequences, impacting policy, investments, and

public opinion. Ensuring data accuracy, therefore, becomes not just a technical challenge, but a moral imperative.

Making Climate Data Easier to Find, Consume, Share, and Trust

The need for a more effective way to manage this deluge of data brings us to the concept of Data Mesh. Data Mesh, an innovative approach to data architecture and organizational design, holds immense potential for transforming the way climate data is handled. This paradigm shift proposes a decentralized approach to data management, focusing on domain-oriented decentralized data ownership and architecture.

Data Mesh presents a forward-thinking approach to addressing the multifaceted challenges inherent in managing climate data. This section delves deeper into how Data Mesh transforms the labyrinth of climate data into a more navigable and efficient system.

The central issue with traditional, centralized data platforms lies in their inherent limitations when dealing with the sheer volume and complexity of climate data. In a standard enterprise setting, these systems often struggle with managing internal data effectively. The climate data challenge amplifies

this complexity exponentially, rendering centralized systems inadequate. In such scenarios, these systems often become overwhelmed, leading to inefficiencies and data silos.

Data Mesh, with its decentralized, domain-oriented architecture, emerges as a more robust and adaptable solution. By distributing data ownership and conceptualizing data as a standalone product, rather than a byproduct of various processes, Data Mesh offers a scalable and responsive framework. This paradigm shift facilitates a network of interconnected data domains, each functioning as a node that enables efficient data sharing and utilization across diverse platforms and sources.

A primary obstacle in harnessing climate data effectively is the disparate nature of its sources. Data Mesh, through its federated approach to data management, addresses this challenge head-on. It establishes a cohesive system that links varied data sources while preserving their individual autonomy. This model is akin to a well-organized library, where books from numerous publishers are readily accessible, yet each retains its distinct identity.

Viewing data as a product under Data Mesh transforms the way data is managed. It assigns clear ownership and responsibility,

paralleling a product owner's role in a company. Owners of climate data domains are thus accountable for the data's accuracy, timeliness, and relevance. This shift not only elevates the quality of climate data but also bolsters user trust, as each dataset is meticulously curated and maintained.

Data Mesh recognizes the need for diverse expertise in managing the broad spectrum of climate data. By allocating specific datasets to domain experts or designated entities, the framework ensures that data is handled by those best equipped to understand and interpret it. This decentralized ownership model not only enhances data accuracy and reliability but also expedites decision-making. Since domain owners are closer to the source of the data, they can implement real-time updates and modifications more effectively. This approach draws inspiration from the "shift-left" concept in software development, incorporating agile methodologies into data management, thereby making the entire system more dynamic and responsive to changes.

Introducing Climate Quantum Inc

To address these challenges, we will use a fictional firm called "Climate Quantum Inc", shown in Figure 3-1, as our case study.

This firm uses Data Mesh to address these challenges and, in turn, makes climate data easy to find, consume, share, and trust.

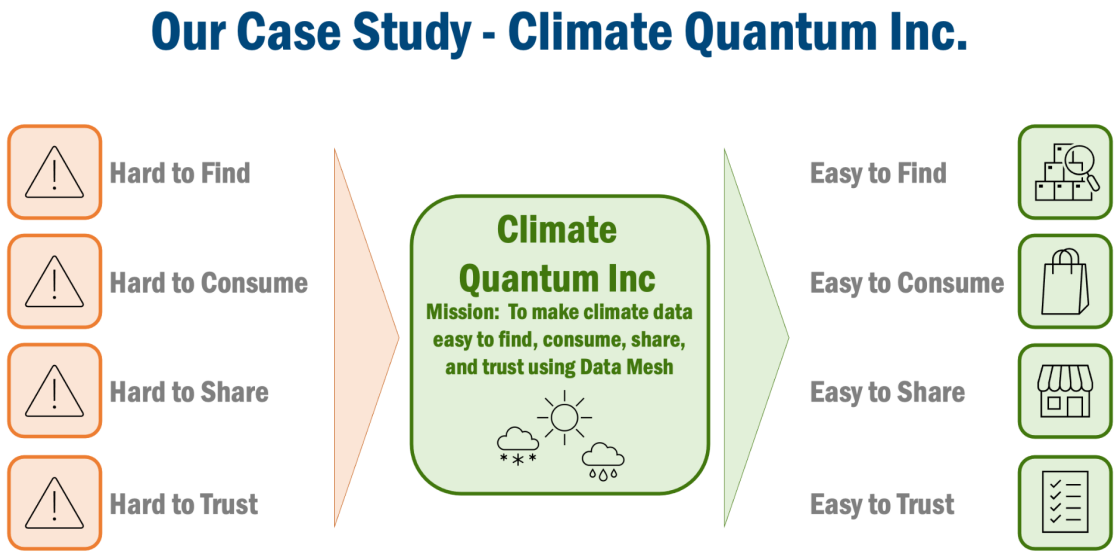


Figure 3-1. :Case Study: Climate Quantum Inc.

Climate Quantum Inc., our hypothetical enterprise, embodies a mission to revolutionize the accessibility, usability, and reliability of climate data by leveraging the Data Mesh framework. This innovative firm stands at the forefront of tackling the multifaceted challenges associated with climate data, streamlining its management, and enhancing its impact.

The mission of Climate Quantum Inc. is multilayered, each aspect addressing a critical facet of climate data management:

Making Climate Data Easy to Find:

One of the most significant hurdles in the realm of climate data is its fragmentation. Vital datasets are dispersed across myriad sources, creating a labyrinthine challenge in locating specific data. This dispersion not only leads to inefficiencies but also creates significant gaps in data availability. Climate Quantum Inc. utilizes Data Mesh's discovery capabilities, including a comprehensive set of APIs and a meticulously curated catalog of data products. This approach transforms the search for climate data into a streamlined, efficient process, ensuring every data product and the information it contains is readily accessible.

Simplifying the Consumption of Climate Data:

The complexity of climate data often stands as a barrier to its effective utilization. Researchers, policymakers, and businesses frequently encounter challenges stemming from inconsistent formats, diverse structures, and limited access. Climate Quantum Inc., through Data Mesh, introduces standardized access methods that demystify the process of consuming climate data. These methods provide clarity and consistency, enabling users to extract valuable insights with greater ease.

Facilitating Easy Sharing of Climate Data:

The sharing of climate data is often impeded by the absence of standardized protocols and overly complex procedures. Traditional centralized models inhibit collaboration, thus hindering the collective response to climate challenges. Climate Quantum Inc. envisions Data Mesh as a solution, offering explicit data contracts that clarify not only the consumption of data but also its sharing. This approach fosters a collaborative environment, essential for effective climate action.

Ensuring the Trustworthiness of Climate Data:

Trust is a cornerstone in the domain of climate data. The reliability of data sources, consistency in quality, and transparency are crucial in building confidence and maximizing the potential of climate data. Through Data Mesh, Climate Quantum Inc. introduces governance certifications that enhance the verification process, ensuring compliance with enterprise and regulatory policies. This framework elevates the trustworthiness of climate data, making it a more reliable and credible resource.

Climate Quantum Inc. is at the vanguard of transforming climate data management through its innovative use of Data Mesh. This comprehensive solution is designed to enhance the discoverability, usability, sharing, and trustworthiness of climate data. The core components of Climate Quantum Inc.'s Data Mesh framework are constructed to address the unique challenges of climate data.

Global Climate Data Mesh:

At the heart of Climate Quantum Inc.'s strategy is a Global Climate Data Mesh. This expansive framework is capable of supporting hundreds of Data Products. It is orchestrated by largely autonomous Data Product teams, each taking responsibility for their specific climate data subsets. This decentralized approach enables these teams to curate, validate, and maintain their data with exceptional efficiency and effectiveness. Such autonomy ensures that the data is not only accurate but also readily accessible.

Climate Data Registry:

The Climate Data Registry functions as a centralized directory for climate data, analogous to the internet's Domain Name System (DNS). It streamlines the process of

finding Data Products, pooling information from diverse global sources. This user-friendly platform demystifies the search for climate data, offering users easy and seamless access to an extensive repository of information.

Data Products: Climate Quantum Inc.

harnesses data from hundreds of climate data sources, transforming them into Data Products. These products represent a vast array of climate-related information, encompassing everything from historical weather patterns to predictive climate models. Each Data Product is a distinct entity, meticulously crafted to provide specific insights and information.

Data Consumption Mechanisms:

To ensure uniformity and ease of access, Climate Quantum Inc. has implemented standardized data contracts. These contracts harmonize data structures and formats, allowing for a more streamlined and coherent user experience. Additionally, the company provides consistent and well-documented APIs and query interfaces, enhancing the ease with which users can interact with and consume climate data.

Data Trust and Verification Mechanisms:

Recognizing the critical importance of trust in climate data, Climate Quantum Inc. has incorporated robust data contracts that rigorously verify and certify the origination, lineage, and quality of the data. The firm has initiated a certification program, akin to standards set by organizations like the American Standards Association, to provide a seal of trust for Data Products that meet stringent quality and transparency criteria. This certification not only ensures data integrity but also reinforces user confidence in the data's reliability.

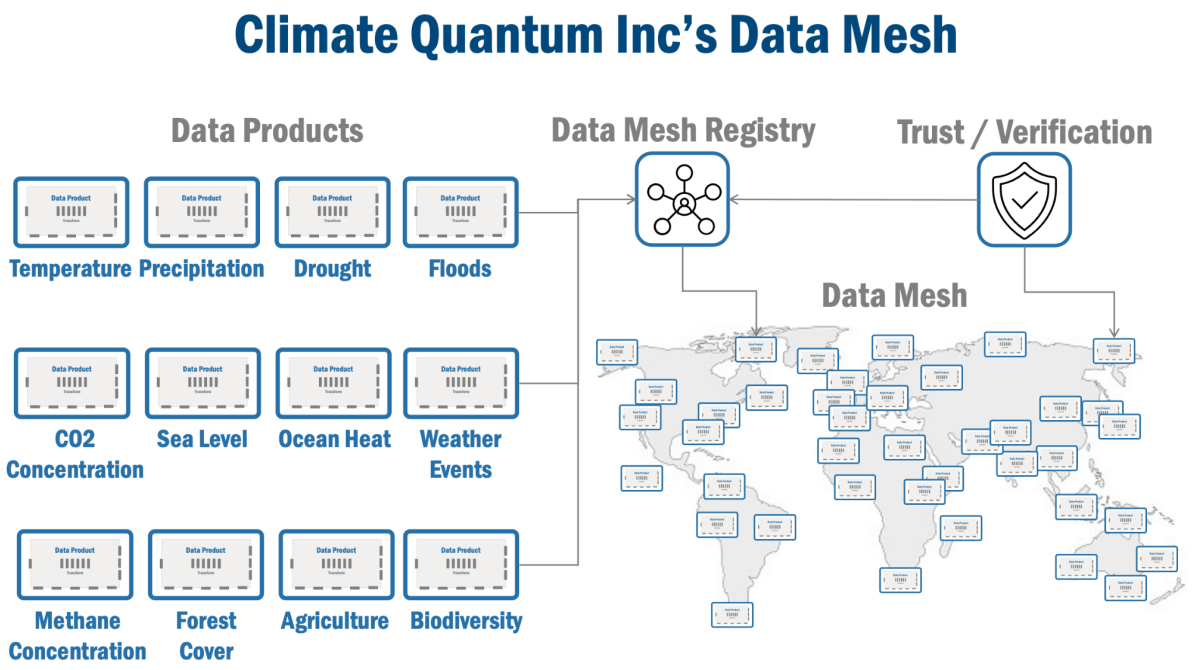


Figure 3-2. : Climate Quantum Inc

Applying Climate Quantum Inc to Your Enterprise

Addressing the challenges faced by Climate Quantum in managing climate data offers valuable insights for enterprises looking to navigate similar complexities within their own data landscapes. The parallels between the climate data challenges and those in a large enterprise are striking, and the strategies employed by Climate Quantum can provide a roadmap for effective data management in any organization.

Firstly, the issue of data fragmentation and discoverability in climate data mirrors the common enterprise challenge of siloed and inaccessible data. Just as Climate Quantum utilized a Global Climate Data Mesh to facilitate data discovery and access, enterprises can adopt a similar Data Mesh approach to break down silos. By implementing a decentralized, domain-oriented data architecture, businesses can ensure that data from various departments and sources is easily discoverable and accessible, enhancing overall efficiency and decision-making capabilities.

Secondly, the complexity and diversity of climate data sources have their counterpart in the varied and often inconsistent data within large enterprises. Climate Quantum’s use of

standardized data contracts and consumption mechanisms to streamline data structures can be replicated in an enterprise setting. By establishing uniform data formats and access protocols, businesses can simplify the consumption of data across different departments, making it more usable and reducing the time and resources spent on data preparation and interpretation.

The challenge of data sharing, crucial in the realm of climate data, is equally pertinent in enterprises where collaboration across departments and with external partners is essential. Climate Quantum's implementation of explicit data contracts to facilitate data sharing is a model that enterprises can emulate. By clearly defining how data can be shared and consumed, enterprises can foster a culture of collaboration and openness, leading to more innovative solutions and a more cohesive organizational approach to data.

Finally, the need for trust and verification in climate data is a universal concern in any data-driven decision-making process. Climate Quantum's approach of employing robust data contracts and a certification program to ensure data quality and transparency can be applied to enterprise data management. Establishing similar governance structures and quality standards in an enterprise will build trust among stakeholders,

ensuring that decisions are based on reliable and verified data. This approach not only enhances the integrity of the data but also reinforces the organization's commitment to data-driven excellence.

In our exploration of the Climate Quantum Inc. case study, we will demonstrate how the principles of Data Mesh can be effectively turned into practice. This will involve a comprehensive examination of the Data Mesh and Data Product architecture, showcasing the core components of the Data Mesh and the individual Data Products it comprises. We will detail the structure, interactions, and contributions of these components, illustrating how data is segmented into distinct, manageable units, each overseen by autonomous teams.

Additionally, we will delve into the organizational design, operating model, and team topologies of the Data Mesh using Climate Quantum as an example. This aspect will focus on the approach to structuring the Data Mesh organization, highlighting how teams are aligned, the roles and responsibilities within these teams, and how they collaborate to maintain the efficacy of the Data Mesh system.

The book will also cover the Data Mesh contracts that facilitate interactions between the various components and Data

Products within the Data Mesh. Climate Quantum data contracts will be used as sample artifacts. These contracts are pivotal in defining how data is shared, accessed, and utilized across different domains, ensuring seamless interoperability and maintaining data integrity.

Finally, we will present a roadmap for building a global climate Data Mesh. This roadmap will outline the step-by-step process for developing and implementing a Data Mesh system on a global scale, tailored specifically to the complex and dynamic nature of climate data. Climate Quantum will be a vehicle that will show strategies for scaling the system, integrating new data sources, and evolving and governing the architecture to meet changing needs and challenges.

Part II. Designing, building, and deploying Data Mesh

With the basic concepts understood, you can now take your journey to designing and building data mesh components. Most of the work might be done by software engineers under the supervision of data engineers & architects. This section will target both groups of engineers (software & data) but will ensure terms that may not be familiar to data engineers are explained.

In chapter 4, you will start by discovering the architecture, which is critical to understanding the different components.

Chapter 5 will dive deeply on data contracts, and why they are essential to creating data products, and later, Data Mesh.

In chapter 6, you will implement your first data quantum (or data product).

Chapter 7 will fly you through the different planes of Data Mesh.

In chapter 8, you will mesh your different data products (or data quanta) to create Data Mesh.

Finally, in chapter 9, you will read more about how AI and Generative AI fits in the Data Mesh model.

Happy reading!

Chapter 4. Defining the Data Mesh Architecture

A NOTE FOR EARLY RELEASE READERS

With Early Release ebooks, you get books in their earliest form—the author's raw and unedited content as they write—so you can take advantage of these technologies long before the official release of these titles.

This will be the 4th chapter of the final book. Please note that the GitHub repo will be made active later on.

If you have comments about how we might improve the content and/or examples in this book, or if you notice missing material within this chapter, please reach out to the editor at sevans@gmail.com.

At the heart of this ecosystem are data products, which form the foundational units of Data Mesh. Each data product in the ecosystem is designed to be discoverable, observable, and operable, ensuring that data can be efficiently shared and utilized across different parts of an organization. Figure 4-1 illustrates the data product architecture which will be

elaborated upon below. There are three key groups of capability within the data product architecture: Definition, Run-time, and Operations.

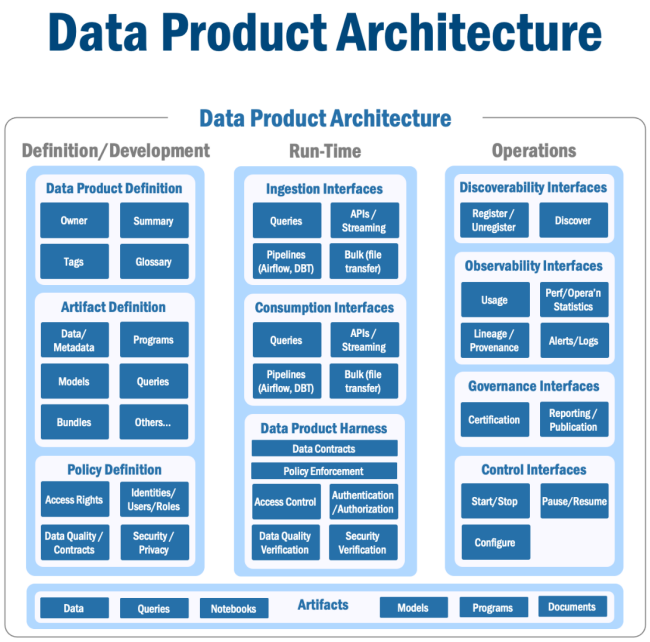


Figure 4-1. : Data Product Architecture

Definition

Data Product Definition is a crucial process in managing and utilizing data effectively. It encompasses the detailed characterization of a data product, including who owns it, what it contains, and the rules governing its use. This clear and comprehensive definition is essential for ensuring that data products are not only functional and accessible but also align

with organizational policies and user needs. By thoroughly defining each data product, organizations can maximize their data's value and utility, making this process a foundational aspect of modern data management strategies.

Defining the core high-level information about a data product begins with identifying the data product owner. This individual plays a pivotal role, acting as the decision-maker and primary source of knowledge for the data product. The owner is responsible for overseeing the development, maintenance, and overall strategy of the data product. They ensure that it meets the evolving needs of its users and stays aligned with the broader goals of the organization. The owner's deep understanding of the data product's capabilities and uses makes them an invaluable resource for answering queries and guiding users in leveraging the data product effectively.

Next, a clear and concise data product summary, along with relevant tags, is essential for users to quickly grasp the essence and scope of the data product, and support effective search capabilities that make it easy to find the data product within Data Mesh. The summary should provide an overview of the data product's purpose, its primary features, and potential applications, enabling users to ascertain its relevance to their needs at a glance. Tags act as keywords or labels, helping

categorize and organize data products within the larger ecosystem. They facilitate easier discovery and retrieval, especially in environments with a multitude of data products. Well-chosen tags can greatly enhance the user experience by simplifying navigation and search processes within the data product landscape.

Last but certainly not least, the creation of a data product glossary is a critical step in fostering a shared understanding among the community that interacts with the data product. This glossary should include definitions of terms and concepts specific to the data product, clarifying any technical jargon or industry-specific language. It serves as a reference tool that ensures consistency in terminology, helping new users acclimate and enabling effective communication among experienced users. A well-constructed glossary not only aids in comprehension but also contributes to building a cohesive community of users who are well-versed in the language and nuances of the data product.

Moving to artifact definition, the term ‘artifact’ is broadly understood in the context of a data product. Artifacts encompass not just traditional data sets but also include any element that a data product owner deems valuable for consumers. This expansive definition reflects the diverse nature

of data usage in contemporary settings, where data is not just stored but also actively manipulated and analyzed.

Artifacts in a data product can include programs and models integrated with the data, providing enhanced functionality and analytical capabilities. These programs and models can be crucial for interpreting the data, drawing insights, and supporting decision-making processes. They add a dynamic aspect to the data product, transforming it from a static repository of information into a versatile tool for analysis and exploration.

Queries that act upon data within the data product are also considered artifacts. These queries can range from simple retrieval operations to complex analytical processes, enabling users to interact with and extract value from the data. The inclusion of queries as artifacts underscores the interactive nature of modern data products, where engagement with data is as important as the data itself.

The concept of artifact definition extends to bundles of data, models, or queries that are tightly integrated. These bundles represent cohesive units of functionality, offering a comprehensive suite of tools and data for specific purposes. Such integration facilitates ease of use, ensuring that users have

access to all necessary components in a harmonized and streamlined manner.

The definition of policies for a data product is a critical component that governs its usage and ensures the security and privacy of the data. These policies delineate the acceptable ways in which the data can be accessed and utilized, taking into account various considerations like access rights, identity management, user authorization, and compliance with security and privacy regulations. Establishing clear and comprehensive policies is essential for maintaining the integrity of the data product and for protecting it from unauthorized access or misuse.

Access rights are the first key consideration in policy definition. They determine who can access the data product and what level of access they are granted. This can range from read-only access for some users to full administrative rights for others. Defining access rights involves assessing the needs and responsibilities of different users or user groups and assigning access levels accordingly. Effective management of access rights is crucial for ensuring that users can perform their roles efficiently while preventing unauthorized access to sensitive data.

Integration with identity books of record is another vital aspect of policy definition. By linking the data product with an enterprise's identity management systems, organizations can streamline the process of user authentication and authorization. This not only enhances security by ensuring that only authenticated users gain access but also simplifies the administrative process of managing user access across various data products and systems.

Defining authorized users and/or roles forms the third cornerstone of policy development. This involves specifying which individuals or roles within the organization are permitted to access the data product and what actions they are authorized to perform. This distinction is crucial for maintaining operational security and for ensuring that each user has access to the appropriate level of data and functionality. By clearly defining authorized users and roles, organizations can maintain tight control over their data and prevent unauthorized use.

Finally, the policy definition must encompass any other security or privacy rules that are necessary to comply with enterprise standards or regulatory imperatives. This includes adherence to industry-specific regulations, data protection laws, and internal security policies. Ensuring compliance with these regulations

and standards is imperative for protecting sensitive data and for maintaining the organization's reputation and legal standing. This aspect of policy definition requires staying abreast of evolving regulatory landscapes and ensuring that data product policies are regularly updated to reflect these changes.

Data Product Harness

Interfaces are implemented using a "harness", which provides a consistent implementation for all interactions with the data product. They also provide the focal point for data contract implementations within a data product. In fact, interfaces and their related data contracts and policy enforcement points work together to allow data products to interact safely, securely, and aligned to expectations as shown in Figure 4-2 below..

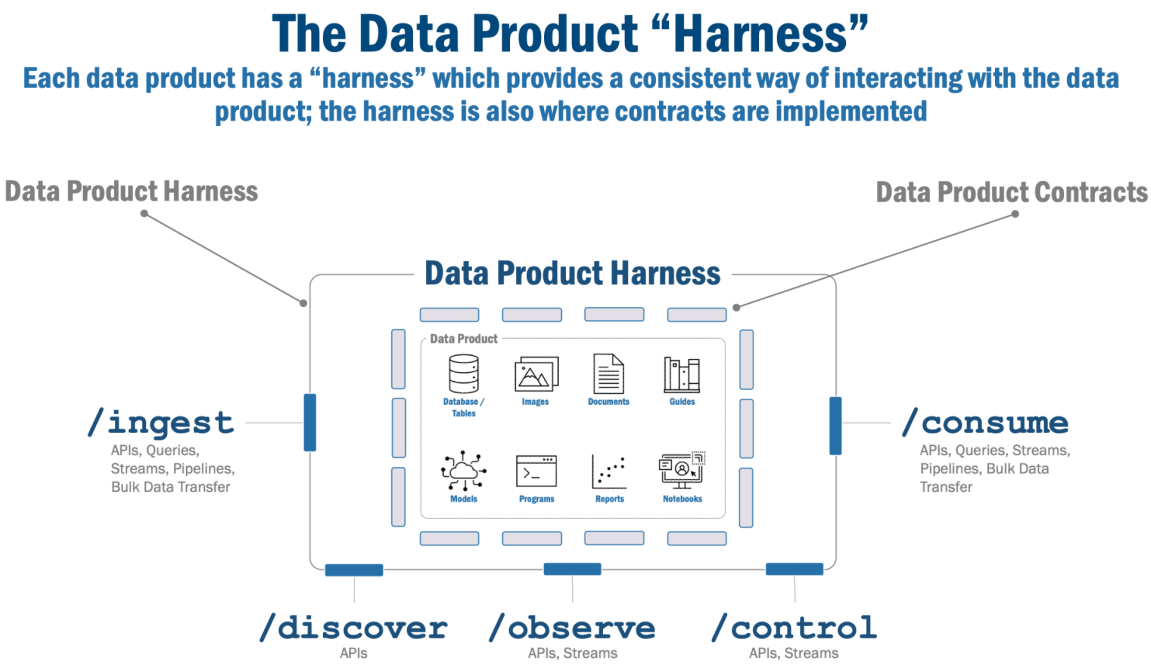


Figure 4-2. : The Data Product Harness

The data product harness is the framework and code that implements the various interfaces for a data product. An architecture goal for any enterprise data mesh is to make these "harnesses" consistent for all data products and thereby making it simpler and more intuitive to interact with any data product in the enterprise data mesh. Now, indeed, every data product is different and embodies unique capabilities, but the mechanisms by which they interact, and the signatures for their interactions can be made largely consistent. Yes, even for ingestion ("/ingest" interface) and consumption ("/consume" interface) - while the "parameters" required for each interaction may be different, but the mechanisms for their interactions can be standardized.

And this is even more practical for other core and foundational interfaces for discovery (“/discover” interface), observability (“/observe” interface), and control or management capabilities (“/control) interface). By doing this, all interactions with data products become consistent and standardized - which means that templating and “factories” become a practical consideration that streamlines and speeds the creation and management of data products.

Now, since all interactions with a data product occur with its “harness”, it makes sense to have each interaction with the data product also be verified and validated against its data contracts. So, the data product harness becomes the integration point for data contracts and associated policy enforcement mechanisms (which is covered in more detail in Chapter X).

Run-Time

The run-time architecture describes the components in a running data product. Let’s dive a bit deeper.

Ingestion Interfaces

Ingesting data into a data product is a fundamental process that determines how data is collected, processed, and made available for use. The method chosen for data ingestion depends on various factors such as the volume of data, frequency of updates, and the specific requirements of the data product. Understanding and selecting the right ingestion method is crucial for ensuring the efficiency and effectiveness of the data product. From APIs to bulk ingestion and pipelines, each technique has its own advantages and ideal use cases.

Queries are a common method for ingesting data, particularly useful when dealing with real-time or near-real-time data updates. They are ideal for situations where data needs to be pulled frequently and in small amounts. Queries allow for specific data to be selected and retrieved based on certain criteria, making them efficient for targeted data ingestion. This method is particularly useful for data products that require up-to-date information and have the capability to handle frequent, incremental updates.

APIs (Application Programming Interfaces) are another popular method for data ingestion, especially effective for smaller datasets or when data needs to be integrated from external sources. APIs facilitate a controlled and secure way to import data, allowing for specific data sets to be accessed and

transferred. They are particularly useful when the data source and the data product need to communicate in a standardized and consistent manner. APIs are also beneficial when dealing with structured data and when the integration needs to be scalable and maintainable over time.

Bulk ingestion methods, such as file transfers, are suited for scenarios where large volumes of data need to be imported into the data product. This method is typically used for initial data loads or periodic updates where a significant amount of data is transferred at once. Bulk ingestion is efficient in terms of resource usage and time, especially when dealing with large datasets that do not require frequent updates. It is often used in conjunction with data warehousing or when integrating historical data into a data product.

Data pipelines, using tools like Airflow or DBT (Data Build Tool), are designed for more complex data ingestion scenarios. These tools allow for the automation of data workflows, enabling the ingestion, transformation, and loading of data in a more controlled and systematic manner. Pipelines are particularly useful when the data ingestion process involves multiple steps, such as data cleansing, transformation, or integration from multiple sources. They provide a robust framework for

managing complex data flows, ensuring consistency and reliability in the data ingestion process.

Other ingestion methods include streaming data ingestion, used for real-time data processing, and web scraping, for extracting data from web sources. Streaming data ingestion is ideal for scenarios where data is continuously generated and needs to be processed in real-time, such as sensor data or user interactions. Web scraping, on the other hand, is useful for extracting data from websites or web applications, especially when other methods of data integration are not available.

Consumption Interfaces

Obviously, once the data has been ingested into the data product, it needs to be consumed. The consumption of a data product involves how end-users access and utilize the data and artifacts provided by the data product owner. It's a process distinct from data ingestion, focusing on the output and user interaction rather than the input of data into the system. While ingestion is about how data gets into the data product, consumption is about how that data, along with other artifacts created by the data product owner, is made available and useful to users. The methods of consumption are diverse, each suited

to different needs and scenarios, ranging from queries and APIs to bulk transfers and pipelines.

Queries are a fundamental method for data consumption, allowing users to retrieve specific subsets of data based on their requirements. This method is particularly useful for users who need immediate and direct access to the data, often for real-time analysis or reporting. Queries enable a high level of flexibility, letting users extract precisely what they need, when they need it. They are ideal for interactive environments where users actively explore and analyze data, such as business intelligence tools or interactive dashboards.

APIs (Application Programming Interfaces) provide a structured and programmatic way to access data. They are particularly effective for consuming smaller amounts of data and are commonly used in applications that require regular, automated retrieval of data. APIs ensure consistency and standardization in data access, making them suitable for integrating data into external applications or services. They are a popular choice for developers building applications that need to interact dynamically with the data product.

Bulk consumption methods, such as file transfers, are used when large volumes of data need to be accessed, often for

offline processing or analysis. This method is typical in scenarios where the entire dataset or large parts of it are required, such as for data warehousing or big data analytics. Bulk consumption is less about real-time interaction and more about comprehensive access, making it suitable for use cases where extensive data processing is necessary.

Data pipelines, utilizing tools like Airflow or DBT, are used for more complex consumption scenarios where data needs to be processed, transformed, or integrated into other systems. These pipelines allow for automated workflows that can handle large volumes of data efficiently. They are particularly useful in scenarios where the data needs to be enriched, aggregated, or transformed before being consumed, ensuring that users have access to high-quality and relevant data.

Beyond raw data, a data product may include a variety of other artifacts for consumption. This can range from pre-defined and vetted queries that facilitate easy access to common data sets, to more complex offerings like Jupyter Notebooks, programs, AI/ML models, and documents. These artifacts add significant value to the data product, allowing users not just to access data, but to interact with it in more meaningful and sophisticated ways. For example, Jupyter Notebooks can provide an interactive environment for data exploration and analysis,

while AI/ML models can offer advanced insights and predictions based on the data.

Policy Enforcement

Enforcing policies in the context of a data product is a critical aspect of ensuring that the data is used appropriately and securely, and, ultimately, is trusted. Policy enforcement goes beyond merely defining what the policies are; it involves implementing mechanisms and controls that actively ensure compliance with these policies during runtime. This is where the theoretical framework of a policy meets the practical aspects of its application. Effective policy enforcement is essential for maintaining data integrity, security, and privacy, and for ensuring that the data product aligns with both organizational standards and regulatory requirements.

Access rights enforcement is the first line of defense in policy implementation. This involves setting up robust authentication and authorization systems to ensure that only authorized users have access to the data product, and only to the extent that their privileges allow. Authentication verifies the identity of a user, typically through credentials like usernames and passwords, while authorization determines the level of access

an authenticated user should have. This might involve role-based access control (RBAC) systems, where different roles within the organization are granted varying levels of access to the data product based on their responsibilities.

Integration with identity books of record is another crucial aspect of policy enforcement. This ensures that user identities and roles are managed centrally and consistently across the organization. By integrating the data product with these central identity management systems, it's possible to streamline the process of granting and revoking access, updating user roles, and monitoring user activity. Such integration not only simplifies the management of user access but also enhances security by maintaining a single source of truth for user identities and permissions.

Enforcing data quality and related data contracts is vital for ensuring that the data within the product remains accurate, consistent, and reliable. This involves implementing checks and validations both at the point of data ingestion and during data usage. Data quality rules might include constraints on data formats, ranges, or the presence of mandatory fields, while data contracts could enforce rules about data relationships and integrity. Automation plays a key role here, with systems set up

to continuously monitor and validate data against these predefined standards and rules.

Security and privacy enforcement is critical, particularly in the context of increasing data breaches and stringent regulatory requirements. This involves a multi-layered approach, including encryption, regular security audits, and adherence to compliance standards such as GDPR or HIPAA. Encryption ensures that data is protected both in transit and at rest, while security audits help in identifying and mitigating vulnerabilities. Compliance with legal and regulatory standards requires a deep understanding of these regulations and the implementation of processes and controls that align with them.

Operations Experience

In the intricate world of data management, particularly within a data mesh framework, the operational considerations for data products encompass a spectrum of interfaces that ensure these products are not only functional but also align with overarching organizational standards and user expectations. These considerations include discoverability, observability, governance, and control interfaces, each playing a pivotal role in the lifecycle and utility of a data product. Discoverability

interfaces enable data products to be registered and located within the data mesh, ensuring that they are visible and accessible to users. Observability interfaces offer insights into the performance and usage of data products, allowing for effective monitoring and management. Governance interfaces are critical for maintaining data integrity and compliance, whereas control interfaces provide essential tools for managing the operational state of data products.

The harmonious interplay of these interfaces is fundamental to the successful operation of data products within a data mesh. Discoverability interfaces facilitate ease of access, making data products easily identifiable and searchable within the vast digital ecosystem. This visibility is crucial for users seeking specific data solutions, ensuring that valuable data products do not remain underutilized or hidden. Observability interfaces, on the other hand, delve into the operational aspects, providing detailed statistics on usage, performance, and even data lineage, which are essential for diagnosing issues and optimizing performance. Governance interfaces allow data product owners to demonstrate their adherence to quality and regulatory standards, instilling confidence among users about the reliability of the data. Lastly, control interfaces empower owners with the ability to directly manage their data products, from simple start/stop operations to more complex

configuration tasks, ensuring that these products remain responsive to the evolving needs of the data mesh and its users.

Discoverability Interfaces

Discoverability interfaces in data products play a crucial role in the data mesh ecosystem, acting as the beacon that signals the presence of a data product to the rest of the system. These interfaces allow data products to register themselves, a process akin to placing a pin on a digital map, marking their location and existence in the Data Mesh landscape. This registration is vital, as it ensures that the data product is not just a standalone entity but a recognized part of the larger network. The information provided during registration typically includes basic details like the data product's name, a brief description, and the identity of its owner, providing a quick but essential overview for potential users.

The depth of discoverability goes beyond simple registration, delving into the detailed attributes of the data product. This includes information about the type of data it contains, its structure, and any specific functionalities it offers. Such detailed disclosure is crucial in an environment teeming with diverse data products, each with its unique characteristics and

capabilities. By thoroughly outlining its attributes, a data product enhances its visibility and usability, guiding users to make informed decisions about which products best suit their needs. This level of transparency is instrumental in fostering a user-friendly ecosystem where data products are easily navigable and accessible.

Furthermore, discoverability interfaces play a significant role in facilitating efficient data product utilization. They help users quickly identify the right data products, reducing the time and effort typically spent in trial and error. This efficiency is particularly important in scenarios where timely access to the right data can significantly impact decision-making processes or operational efficiency. By streamlining the discovery process, these interfaces ensure that the value of the data mesh is maximized, benefiting both data product providers and consumers.

Observability Interfaces

Observability interfaces within data products play a crucial role in ensuring that these resources are not just operational, but also efficient, reliable, and transparent in their functioning. These interfaces provide a comprehensive view into the various

operational aspects of a data product, including usage statistics, performance metrics, and overall operating health. By offering real-time insights into how a data product is performing, observability interfaces allow data product owners and users to monitor and evaluate the product's effectiveness continuously. This visibility is key to maintaining high performance, as it enables quick identification and resolution of any issues that may arise.

One of the most significant benefits of observability interfaces is their ability to track and report on usage patterns. This includes how often the data product is accessed, which parts are most frequently used, and by whom. Such insights are invaluable for understanding user behavior and preferences, allowing for informed decisions on future enhancements or modifications to the data product. This data-driven approach ensures that the product evolves in line with user needs and remains relevant and useful over time. Moreover, usage statistics can also aid in resource allocation and scaling decisions, ensuring optimal performance even as demand fluctuates.

In addition to usage statistics, observability interfaces also focus on performance and operating statistics. This encompasses everything from load times and response rates to more complex

metrics like data throughput and processing efficiency. Monitoring these aspects is crucial for maintaining a high level of service quality and for preemptively identifying potential bottlenecks or performance issues. In today's fast-paced digital environment, where data is a critical asset for decision-making, ensuring that data products perform reliably and efficiently is paramount.

Lastly, the importance of lineage and provenance tracking in observability interfaces cannot be overstated. This feature provides detailed information about the origin and historical changes of the data, which is essential for maintaining data integrity and trust. It ensures transparency in how data has been collected, processed, and transformed over time. Additionally, robust alerting and logging capabilities are essential for problem diagnosis, enabling quick and effective troubleshooting. This not only minimizes downtime but also ensures that data quality and accuracy are maintained, which is critical for making informed decisions based on the data.

Governance Interfaces

Governance interfaces within the realm of data products are vital tools that enable data product owners to manage and

demonstrate the quality and compliance of their offerings. These interfaces are designed to facilitate the process of certification or verification, whereby owners can assert that their data products meet specific data quality standards and expectations. This process is not merely a formality but a crucial aspect of establishing and maintaining the trustworthiness and reliability of the data product. In a landscape where data accuracy, integrity, and compliance are paramount, governance interfaces provide the means to ensure and showcase these qualities.

The importance of governance interfaces extends beyond mere certification. They enable data product owners to publish and report on their governance posture, offering transparency into the data management practices and standards adhered to by the data product. This openness is particularly important in an era where data handling and privacy concerns are at the forefront of users' minds. By providing clear insights into their governance practices, data product owners can build confidence among users, reassuring them that the data is managed responsibly and in accordance with best practices and regulatory requirements.

Furthermore, governance interfaces play a crucial role in aligning data products with organizational and regulatory

standards. In an environment where data regulations are becoming increasingly stringent, these interfaces help ensure that data products are not only compliant with current regulations but are also prepared to adapt to future changes. This adaptability is key in a dynamic regulatory landscape, enabling data product owners to swiftly adjust their governance practices to meet evolving requirements. Consequently, governance interfaces are instrumental in mitigating risks associated with non-compliance, such as legal penalties or reputational damage.

Control Interfaces

Control interfaces in the context of data products are essential tools that provide data product owners with the ability to manage the operational state of their offerings. These interfaces encompass a range of functionalities, including the capability to start or stop the data product, pause or resume its operations, and configure various settings and parameters. This level of control is crucial for ensuring that data products can be managed effectively and can respond flexibly to different operational requirements or situations. In essence, control interfaces act as the command center for data product owners,

offering them direct oversight and management capabilities over their products.

The importance of these control interfaces becomes particularly evident in dynamic operational environments. The ability to start or stop a data product enables owners to manage resource allocation efficiently, ensuring that the data product is active only when needed, thereby optimizing resource utilization and reducing unnecessary costs. Similarly, the pause and resume functionalities are vital in scenarios where temporary interruptions are required, such as during maintenance, updates, or in response to unforeseen issues. These capabilities ensure minimal disruption to the end-users while allowing for necessary adjustments or repairs to be made.

Moreover, the configuration aspect of control interfaces plays a significant role in tailoring the data product to specific needs or conditions. Owners can adjust settings to optimize performance, customize features according to user feedback, or adapt to changing data environments. This flexibility is key to maintaining the relevance and effectiveness of the data product over time. In a landscape where user needs and technological environments are constantly evolving, the ability to configure and reconfigure data products swiftly ensures that they

continue to deliver value and meet the expectations of their users.

Data Product Artifacts

The name “data products” suggests that data is an obvious primary consideration. But, importantly, it is not the only consideration. In fact, a much more expansive perspective is offered, where we introduce “artifacts”. Artifacts are not just data but any other object that a data product owner wishes to make available for consumption.

Artifacts, ranging from basic data sets to complex programs and models, add significant value to the data product, making it more than just a data store. They enable a more integrated and user-friendly approach to data management, where the focus is not just on providing data but on delivering a complete, valuable, and ready-to-use data solution. This holistic approach is what sets modern data products apart, making them indispensable tools in the data-driven world.

These artifacts represent a broad and inclusive array of elements that extend well beyond traditional data sets. They embody any set of data or related objects that a data product owner deems valuable for users and wishes to make available

for consumption. This expansive view of what constitutes an artifact reflects a modern approach to data management, where the value of a data product is significantly enhanced by the versatility and comprehensiveness of its components.

At the most basic level, data product artifacts include conventional data elements like databases, tables, and files. These fundamental components form the backbone of any data product, providing the core data that users seek for various applications. In a traditional sense, these data sets are what one might typically expect to find within a data product. However, the scope of artifacts in the Data Mesh concept goes far beyond these basic elements, embracing a more holistic and integrated approach to data management.

Beyond these standard data components, artifacts in a data product can also include programs that are directly related to or integrated with the data. These programs can range from simple scripts that facilitate data processing to complex software applications that provide advanced analytics capabilities. The integration of such programs with the data enhances the overall value of the data product, enabling users to not only access data but also to process and analyze it within the same environment. This integration represents a shift from

static data repositories to dynamic, interactive platforms that offer greater utility and efficiency.

Models, particularly those in the realm of artificial intelligence and machine learning, are another category of artifacts that can significantly enhance the value of a data product. These models can be used to derive insights, make predictions, or uncover patterns within the data. By including AI and ML models as artifacts, data product owners empower users to leverage advanced analytical techniques, thereby expanding the potential applications and value of the data product.

Queries are yet another type of artifact that can be included in a data product. Pre-defined queries or query templates can significantly ease the process of data extraction and analysis for users. These queries can be tailored to common use cases or specific data analysis tasks, enabling users to quickly and efficiently access the data they need. The inclusion of such queries not only makes the data product more user-friendly but also reduces the time and effort required to derive value from the data.

Data product artifacts can also encompass more comprehensive bundles that integrate data with models, programs, or queries. These bundles provide a packaged solution to users, combining

all the necessary components for specific data applications or analyses. This approach simplifies the user experience by providing a ready-to-use set of tools and data, thereby enhancing the efficiency and effectiveness of the data product.

Data Contracts

Data contracts are a fundamental concept in data management, particularly in environments where data sharing and interoperability are key, such as in a Data Mesh architecture. They serve as formal agreements that define the specifics of how data is structured, accessed, and used within and between different data products and systems. These contracts are crucial for ensuring consistency, reliability, and security in data exchanges. Let's break down the key aspects of data contracts:

Structure and Format of Data

Data contracts specify the format and structure of the data. This includes the data types, schemas, and layout that the data adheres to. By defining these parameters, data contracts ensure that when data is shared or transferred between systems, it is understood and interpreted correctly by all parties involved.

Access and Usage Guidelines

Data contracts outline how data can be accessed and used. This might include specifying API endpoints for data access, defining what operations (like read, write, update, delete) are allowed, and setting out any other usage restrictions. This helps in managing data access rights and maintaining data integrity across different users and applications.

Data Quality and Integrity

They often include provisions for data quality, ensuring that the data meets certain standards of accuracy, completeness, and consistency. This is crucial for maintaining the trustworthiness of data, especially when it is used for decision-making or analytical purposes.

Security and Privacy

Data contracts address security and privacy considerations, outlining measures for data protection, encryption, and compliance with data privacy laws like GDPR or HIPAA. This is critical in safeguarding sensitive data and adhering to legal and regulatory standards.

Version Control and Updates

They may also include details on version control and how updates to the data or the contract itself are managed. This ensures that all parties are working with the most current and accurate data and that changes are tracked and communicated effectively.

Data Lineage and Provenance

Some data contracts go a step further to include information about data lineage and provenance, providing transparency about where the data originated and how it has been transformed over time.

Error Handling and Resolution Mechanisms

Finally, data contracts can include protocols for error handling and conflict resolution, outlining the steps to be taken in case of discrepancies or data issues.

So, why do we include data contracts in this architecture chapter? Yes, there will be dedicated chapters that address data contracts, but the simple reason is that they are a foundational concept and, like any foundational concept, applying them after-the-fact is expensive and impractive. Hence we are introducing the concept here, and ensuring that data contracts are addressed explicitly in our data product architecture.

Data Product Governance

Data governance is the set of practices and policies that ensure high-quality, secure, and efficiently used data within an organization. It involves creating rules and processes to manage data accessibility, accuracy, consistency, and security. The goal is to ensure that data is reliable and can be trusted for making important decisions. This involves setting standards for how data is collected, stored, used, and deleted, ensuring it is handled consistently across the entire organization.

An essential part of data governance is ensuring data security and compliance with relevant laws and regulations. This includes protecting sensitive information from unauthorized access and breaches, and ensuring that the organization's data practices adhere to legal standards such as GDPR for privacy. It also involves defining clear roles and responsibilities for those who handle and manage data, ensuring everyone knows who is accountable for different aspects of data management. However, in many organizations, data governance is the responsibility of a central group that defines and enforces enterprise-wide policies.

We would like to offer an alternative to this traditional data governance model. The concept of data governance in the

realm of data products borrows a page from established practices in other industries, such as the American National Standards Institution (ANSI) approach to product certification. In this model, governance is not solely a top-down, centralized affair; rather, it's a federated process where responsibility and authority are distributed among various stakeholders. This paradigm shift is particularly relevant in the context of data products, where empowering data product owners to manage and enforce governance policies can lead to more efficient and responsive governance practices.

In traditional data governance, a centralized group typically sets and enforces policies across the entire organization. This approach, while providing uniformity, can often be slow to adapt and may not adequately address the unique needs of different data products and their consumers.

However, by adopting a federated model, similar to the ANSI product certification process, data governance can be tailored more effectively to the specific requirements of each data product while still aligning with overarching organizational policies.

The process used by the ANSI for governing product quality, safety, and reliability is a blend of centralized and federated

approaches. The process begins with the identification of a need for standardization. Once a need is identified, ANSI facilitates the creation of a consensus body which is responsible for the drafting of the standard.

ANSI's role in the certification process is primarily in the accreditation of third-party certification bodies. These are independent organizations that have the authority to test and certify products against the relevant standards. ANSI's accreditation process ensures that these certification bodies have the necessary competence, consistency, and impartiality to conduct certification effectively. This process involves a thorough evaluation of the certification body's procedures, staff qualifications, and quality control measures.

The actual certification process is carried out by these accredited certification bodies. A manufacturer seeking certification for a product must submit it to one of these bodies. The certification body then tests and evaluates the product to determine if it meets the relevant ANSI or other recognized standards. This evaluation can include a variety of tests, inspections, and assessments, depending on the nature of the product and the standards it must adhere to.

If the product meets the standards, the certification body issues a certificate and allows the manufacturer to label the product as compliant. This certification is a mark of quality and reliability that can be used in marketing and product documentation. Additionally, to maintain certification, manufacturers are typically required to undergo regular follow-up assessments and audits. This ensures that their products continue to meet the necessary standards over time.

Now, how does this apply to data governance, and what role do data products and their owners play?

Under a federated data governance model, centralized data governance acts like the ASA/ANSI: it is responsible for facilitating the definition of standards wherever they may be needed (obviously, there is a prioritization mechanism). In this model, data product owners take on a role akin to product manufacturers in the ASA/ANSI's certification process. They are responsible for ensuring that their data products adhere to both the local and enterprise-level standards, policies, and data contracts. This responsibility includes not only the creation and maintenance of the data product but also the ongoing verification of its compliance with the set standards.

Once a data product owner has verified that their product meets the required standards, they can then “certify” their data product, much like a product earning the right to display the ASA logo. This certification is a public declaration that the data product has been vetted and meets the prescribed criteria for quality, safety, and reliability. This process encourages a culture of accountability and transparency among data product owners, as their products’ compliance is openly demonstrated and can be scrutinized by users and stakeholders.

The governance interfaces, or APIs, in this system play a crucial role, much like the ASA’s oversight in the product certification process. These interfaces allow anyone within the organization to query a data product’s governance status at any time. This feature ensures that the governance information is not just a static badge but a dynamic and continually up-to-date reflection of the data product’s compliance with the established standards.

This approach to data governance, inspired by the ASA/ANSI model, offers several advantages. It allows for more rapid adaptation and response to changing needs and conditions within different areas of the organization. Since data product owners are closer to the specifics of their products and their

consumers' needs, they are better positioned to implement relevant and effective governance measures.

Moreover, this model fosters a more engaged and responsible approach among data product owners. Just as manufacturers strive to maintain their product certifications for market credibility, data product owners are incentivized to uphold their governance standards to maintain the trust and confidence of their users. This shift from a centralized governance model to a more federated approach mirrors the evolution in other industries, where empowering local stakeholders leads to higher standards and greater innovation.

So, why is this section discussed in the architecture section? For one simple reason - recognizing it is much different than traditional models, it may not be practical nor feasible to establish a federated data governance process without explicit architecture support to make the process simple, consistent, and effective. And, to wit, this is why we explicitly call out "governance interfaces" within our data product architecture.

Data Mesh Architecture

At its Data Mesh, it is an ecosystem composed of numerous data products, each functioning as a distinct unit with its own

purpose and capabilities. However, what makes Data Mesh truly powerful is its connective tissue – the framework and infrastructure that bind these individual data products into a cohesive and integrated whole. This connective tissue is not just about linking data; it's about creating a synergistic environment where the sum is greater than its individual parts. Each data product in the mesh maintains its autonomy, yet contributes to and benefits from the larger data ecosystem, fostering a dynamic and collaborative data landscape as you see ion Figure 4-3 below.

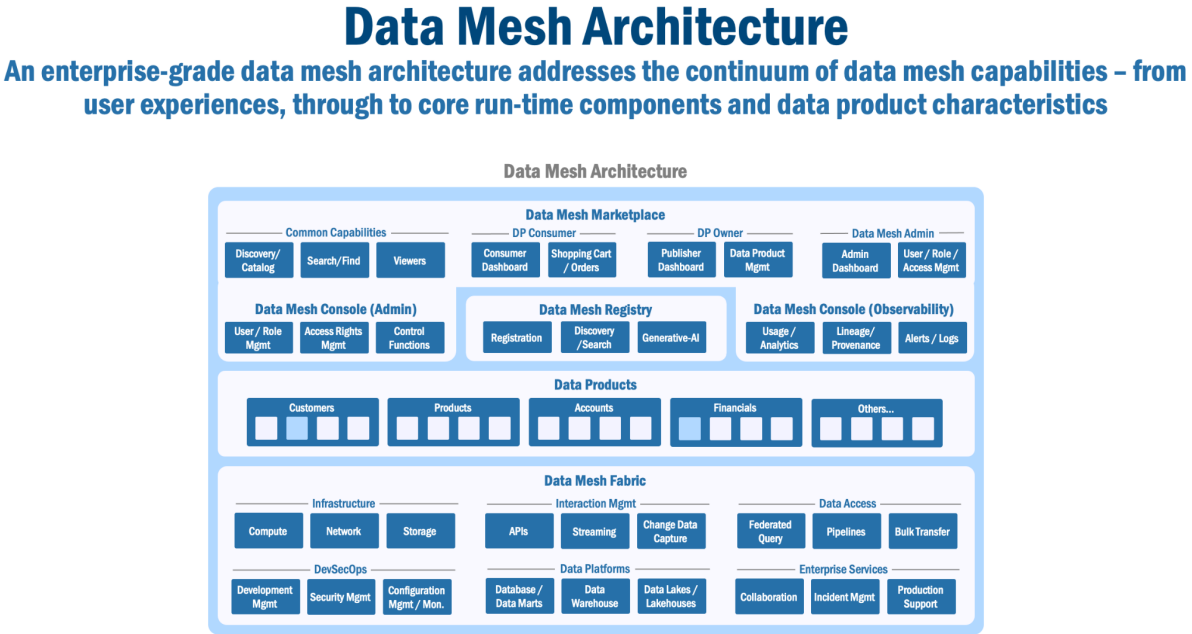


Figure 4-3. : Data Mesh Architecture

Central to the Data Mesh concept is the presence of a marketplace – a platform that simplifies the process of finding,

consuming, sharing, and trusting data products. This marketplace serves as a hub where data products are cataloged, making them easily discoverable and accessible to users across the organization. It allows for efficient sharing and collaboration, enabling users to leverage a wide range of data resources. Trust is also a key component here, as the marketplace provides mechanisms to ensure that the data products meet certain quality and compliance standards. This marketplace functionality is crucial in breaking down data silos, encouraging a culture of open data exchange and collaboration.

Additionally, Data Mesh includes a robust communications backbone or fabric, which facilitates interaction between various data products. This backbone supports a range of data interactions, from API integrations and streaming to pipelines and bulk data transfers. While common platforms for managing these interactions may be offered within the Data Mesh, they are not mandated, providing flexibility for data product owners to choose the best tools and technologies for their specific needs. This flexibility is a hallmark of the Data Mesh approach, allowing for a diverse range of data products and technologies to coexist and interoperate within the same ecosystem, driving innovation and efficiency in data management practices.

Data Mesh Marketplace

The Data Mesh marketplace is a sophisticated user interface designed to revolutionize how data products, along with their data and artifacts, are accessed, utilized, and managed within an organization. It stands as the epicenter of the Data Mesh ecosystem, providing a user-friendly platform that simplifies the process of finding, consuming, sharing, and establishing trust in data products. This marketplace is not just a repository; it's a window into a Data Mesh, and an interactive space where data becomes more accessible and actionable for everyone involved, from producers to consumers.

For consumers of data products, the marketplace offers a catalog user experience (UX) that makes it easy to browse and discover available data products. This catalog is designed to be intuitive and user-friendly, ensuring that even those with minimal technical expertise can navigate and find the data products they need. The inclusion of search and find capabilities, utilizing both keyword and natural language search, further enhances this experience. Users can effortlessly search for data products that meet their specific requirements, making the discovery process both efficient and effective.

For producers who provide data to these products, the marketplace offers valuable tools to manage and monitor their offerings. This includes dashboards that display commonly consumed or viewed data products and their usage patterns. These insights are crucial for producers to understand the impact and reach of their data products, allowing them to make informed decisions on how to evolve and improve their offerings.

The marketplace also addresses the practicalities of data consumption, especially when dealing with large volumes of data. It features shopping carts and order systems where direct downloading of data might not be practical. This asynchronous delivery system is particularly useful for handling large data sets, allowing users to “order” the data they need and receive it in a manageable and efficient manner.

For data product owners, the marketplace provides a more detailed dashboard. This feature offers deep insights into usage patterns and governance of their data products. Such detailed analytics enable owners to understand how their data products are being used, which parts are most popular, and how they can improve their offerings. This level of detail is vital for the continuous improvement and relevance of the data products within the Data Mesh.

On the governance front, the marketplace facilitates a transparent and controlled environment. Data product owners can monitor and enforce compliance with data standards and policies, ensuring that their products meet the required quality and security benchmarks. This aspect of the marketplace is crucial for maintaining trust and integrity in the data products offered.

The admin user experience within the marketplace is another critical component. It allows administrators to establish overarching Data Mesh policies and manage user access and permissions. This central control ensures that the data ecosystem adheres to organizational standards and regulatory requirements while providing flexibility to cater to individual data product needs.

Moreover, the marketplace is designed to foster a culture of data sharing and collaboration. By making data products easily accessible and consumable, it encourages departments and teams within an organization to leverage shared data resources, breaking down silos and fostering a more integrated approach to data utilization.

Additionally, the marketplace includes features to facilitate trust among its users. By providing transparent and detailed

information about each data product, including its source, quality, and compliance status, users can make more informed decisions about which data products to trust and use for their specific purposes.

The marketplace also evolves with user feedback and changing data landscapes. It continuously adapts to meet the emerging needs of its users, whether it's through updating its interface for better user experience or integrating new functionalities to handle different types of data products and artifacts.

Data Mesh Registry

The Data Product Registry within a data mesh ecosystem functions much like the Domain Name System (DNS) does for the internet. It's a streamlined and efficient directory that provides basic yet essential information about the various data products available within the data mesh. This Registry is designed to facilitate easy discovery of data products, acting as the foundational layer that connects users to the data they seek. It maintains only the most crucial information – data product summaries and tags – to assist users in finding the right data products through simple keyword and natural language searches. This design mirrors the DNS's role in the internet,

where it serves as a phone book, directing users to the right web addresses with minimal yet crucial information.

The process of populating the Data Product Registry is intentionally made straightforward and user-friendly. Data product owners are responsible for registering their products, a task that requires them to provide a concise summary, limited to a couple of paragraphs, and a set of relevant tags. The emphasis here is on simplicity and ease of use. The design of the Registry is such that even junior staff members can complete the registration process without difficulty. This approach ensures that the barrier to entry for registering data products is low, encouraging comprehensive participation from all parts of the organization.

Publishing a data product in the Registry is equally streamlined. Whether using a simple user interface or a command-line interface, data product owners can quickly make their data products available in the marketplace. The process is designed to be unobtrusive and efficient, avoiding any unnecessary complexity or technical hurdles. This ease of publishing ensures that the Registry remains up-to-date and reflective of the current data offerings within the data mesh.

Once a data product is registered, it becomes immediately available in the data marketplace. This rapid availability is crucial in maintaining the dynamism and responsiveness of the data mesh. Users can access newly registered data products without delay, ensuring that the data ecosystem is continuously evolving and expanding. This immediate availability aligns with the overarching goal of the data mesh to provide timely and easy access to a wide range of data resources.

For users navigating the marketplace, the Registry serves as their first point of interaction with the data products. Upon conducting a search, users are presented with links to the data products that match their criteria. These links are derived from the Registry's concise summaries and tags, providing just enough information to guide users to the right data product. This user experience is akin to using a search engine on the internet, where the search results lead to various web pages based on the entered keywords.

Once a user selects a data product from the marketplace, they are then directed to more detailed interfaces of the data product, such as 'discovery', 'observability', or 'governance'. These interfaces provide deeper insights into the data product, allowing users to understand its structure, usage patterns, compliance status, and more. This two-tiered approach –

starting with the simplicity of the Registry and moving to the more detailed interfaces – ensures a balanced user experience, combining ease of discovery with depth of information.

Data Mesh Console

The Data Mesh Console, a critical component of the Data Mesh ecosystem, serves as a command line interface (CLI) that provides comprehensive management capabilities across the entire data mesh. This console is designed for users who prefer or require a more direct, script-based interaction with the data mesh, as opposed to the graphical user interface of the marketplace. It offers similar functionalities to the marketplace but in a format that is more aligned with traditional command line operations. For instance, through the CLI, users can interact with the Data Mesh Registry, accessing and managing data product information efficiently. This command line approach is particularly appealing to those who seek a faster, more streamlined way to navigate and manipulate the data mesh, especially for tasks that are repetitive or require automation.

In addition to registry interaction, the Data Mesh Console provides robust administrative capabilities, crucial for managing the broader aspects of the data mesh. This includes user management, where administrators can add, remove, or

modify user accounts, as well as role and access management, ensuring that the right individuals have the appropriate level of access to various data products. This aspect of the console is vital for maintaining the security and integrity of the data mesh, as it allows for precise control over who can access what data, and under what conditions. The console's administrative functions are essential for enforcing data governance policies and for ensuring that the data mesh remains compliant with organizational standards and regulatory requirements.

The console also excels in offering observability capabilities, providing insights into the usage and performance of the data mesh. Administrators and data product owners can use the CLI to monitor how data products are being used, track performance metrics, and identify any potential issues. Additionally, the console provides tools for examining cross-data product concerns such as data lineage and provenance. This is particularly important for understanding the journey of data across different products within the mesh, ensuring data integrity and compliance with data governance standards. Overall, the Data Mesh Console is a powerful tool for those who manage and oversee the data mesh, offering a range of functionalities that ensure efficient operation, robust security, and insightful oversight of the data ecosystem.

Data Mesh Fabric

The Data Mesh fabric is an integral component of the Data Mesh architecture, serving as the backbone and connective tissue that integrates various data products into a cohesive whole. This fabric is not just a single entity but a conglomeration of various services and platforms, each playing a vital role in ensuring that the data mesh functions seamlessly and efficiently. It's the infrastructure upon which the entire data mesh operates, providing the necessary foundation for data products to communicate, interact, and deliver value.

At the heart of the Data Mesh fabric are the infrastructure services, which include compute, network, and storage capabilities. These services form the core platform that supports almost all components of the data mesh. They provide the essential resources required for data processing and storage, ensuring that data products have the necessary computational power and space to operate effectively. The robustness and scalability of these infrastructure services are crucial for the smooth functioning of the data mesh, especially as the number and complexity of data products grow.

Interaction and communication management services are another critical aspect of the Data Mesh fabric. These services

encompass APIs, streaming technologies, and change data capture mechanisms. They facilitate the interaction and exchange of information between different data products, enabling them to communicate and share data in real-time. This interconnectedness is essential for creating a dynamic and responsive data ecosystem, where data can flow freely and securely between different products and systems.

Data access services form another layer of the Data Mesh fabric. This includes tools and technologies for federated query, data pipelines, and bulk data transfer. These services allow for efficient and flexible access to data across the mesh, regardless of where it resides. Federated query capabilities, for instance, enable users to retrieve and combine data from multiple sources without moving the data, while pipelines and bulk transfer mechanisms provide efficient ways to move large volumes of data when necessary.

DevSecOps, integrating development, security, and operations management, is a cornerstone of the Data Mesh fabric. This component ensures that all services within the data mesh are managed consistently, safely, and reliably. DevSecOps practices embed security considerations into the development lifecycle of data products and services, ensuring that security is not an afterthought but an integral part of the entire process. This

approach is crucial for maintaining the integrity and security of the data mesh, especially in complex and fast-evolving technological landscapes.

Data platforms are another vital component of the Data Mesh fabric. This includes a range of storage and processing solutions like databases, data marts, data warehouses, data lakes, and data lakehouses. Each of these platforms serves a specific purpose, from structured data storage in databases and data marts to large-scale data storage and processing in data lakes and lakehouses. The availability of diverse data platforms within the fabric allows for greater flexibility and choice in how data is stored, processed, and accessed, catering to the diverse needs of different data products.

Collaboration services within the Data Mesh fabric play a key role in fostering a community-centric environment. Knowledge management tools, forums, threads, and social features like likes and comments enable developers, data scientists, and other stakeholders to share insights, ask questions, and collaborate on data projects. These collaborative tools not only enhance the user experience but also drive innovation and knowledge sharing across the data mesh.

Data Product Actors

In a Data Mesh ecosystem, various actors play pivotal roles in ensuring the smooth functioning and effectiveness of the system. These roles can be broadly categorized into five key groups: data product owners, data producers, data consumers, data mesh admins, and data governance professionals. Each group has its own unique set of responsibilities and contributions, making them essential components of the Data Mesh architecture. Understanding the nuances of these roles is crucial for the successful implementation and operation of a Data Mesh, as it ensures that all aspects of data handling, from production to consumption, are effectively managed.

- Data Product Owners are at the forefront of the Data Mesh, overseeing the lifecycle of data products. They are responsible for defining the vision, strategy, and functionality of their data products. This group ensures that the data products align with business objectives and user needs, managing everything from data sourcing to the presentation of data. Data Product Owners play a crucial role in bridging the gap between technical capabilities and business requirements, making sure that the data products deliver real value to the organization.
- Data Producers include operational source systems, other analytics data sources, and even other data products within the Data Mesh. This group is responsible for providing the

raw data that is ingested by various data products. Their role is crucial in ensuring that high-quality, relevant, and timely data is available for further processing and analysis. Data Producers are the foundation of the Data Mesh, as the quality and reliability of their output directly impact the effectiveness of the data products.

- Data Consumers encompass business users, data scientists, and analysts who utilize the data and artifacts contained within data products. They rely on data products to gain insights, make informed decisions, and drive business strategies. This group is the end-user of the Data Mesh, and their feedback often shapes the evolution of data products. Their interaction with the data products is a key indicator of the effectiveness and relevance of the Data Mesh in addressing business needs.
- Data Mesh Administrators are responsible for the overall management and control of the Data Mesh infrastructure. They handle the technical aspects, including system performance, resource allocation, and ensuring the smooth operation of the Data Mesh. This group plays a critical role in maintaining the health of the Data Mesh, ensuring that it is scalable, secure, and efficient.
- Data Governance Professionals facilitate policy management and the data product “certification” process

within the Data Mesh. They ensure that data products adhere to defined standards, compliance requirements, and quality benchmarks. This group is pivotal in instilling a culture of accountability and trust in the Data Mesh, ensuring that data is handled responsibly and ethically across the organization.

Climate Quantum Use Case

So, at this point we understand the architecture of a data product, how it defines its data and artifacts, its ingestion and consumption capabilities, as well its the core operational interfaces. And we have also seen the constituent components of the data mesh and how it binds data products into a broader ecosystem.

Now, let’s see how we can put the pieces together for our Climate Quantum, our use case. As you recall, Climate Quantum’s mission is to make climate data easy to find, consume, share, and trust using data products and a broader data mesh ecosystem.

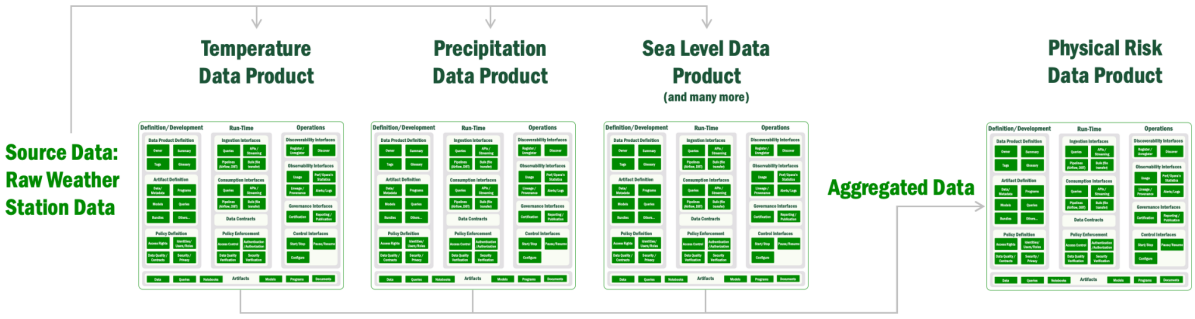


Figure 4-4. : Climate Quantum Data Products

As we know, Climate Quantum is a pioneering initiative in the realm of climate change analysis, leveraging the power of data to assess physical risks due to environmental changes. At the heart of this venture are its meticulously structured data products, each serving a distinct but interconnected purpose. The first three data products – focusing on temperature, precipitation, and sea level – form the foundational layers of this innovative framework. These products are internal but critical components, each aggregating and refining raw weather data to create comprehensive datasets that are both reliable and insightful.

The temperature data product is a key element of Climate Quantum's arsenal. It aggregates vast amounts of raw temperature data from multiple sources, processing raw weather station information to offer a clear and comprehensive view of temperature trends and anomalies. This data product not only gathers information but also intelligently fills in gaps in the data, ensuring that the resulting dataset is as complete and accurate as possible.

Similarly, the precipitation data product is dedicated to capturing and aggregating rainfall and other forms of precipitation. This data product delves into vast datasets, aggregating information to paint a detailed picture of precipitation patterns across different regions and locations. Like its temperature counterpart, this product also focuses on identifying and compensating for any missing data, thereby ensuring a high-quality and comprehensive dataset. The empowered owner of this data product plays a crucial role in maintaining its quality and ensuring that it effectively contributes to Climate Quantum's overall mission.

The sea level data product complements the temperature and precipitation products by focusing on changes in sea levels – a critical aspect of climate change studies. By aggregating data from various sources, this product provides invaluable insights

into sea level trends, which are essential for understanding the long-term impacts of climate change on coastal areas. The owner of this data product is responsible for its accuracy, completeness, and ongoing relevance, making it a vital component of the Climate Quantum framework.

The culmination of Climate Quantum's data products is the physical risk data product. This data product synthesizes the aggregated data from the temperature, precipitation, and sea level products and applies advanced AI/machine learning techniques to model the physical risks associated with climate change at specific locations. This model offers critical insights to the public, enabling governments, businesses, and communities to understand and prepare for the potential impacts of climate change.

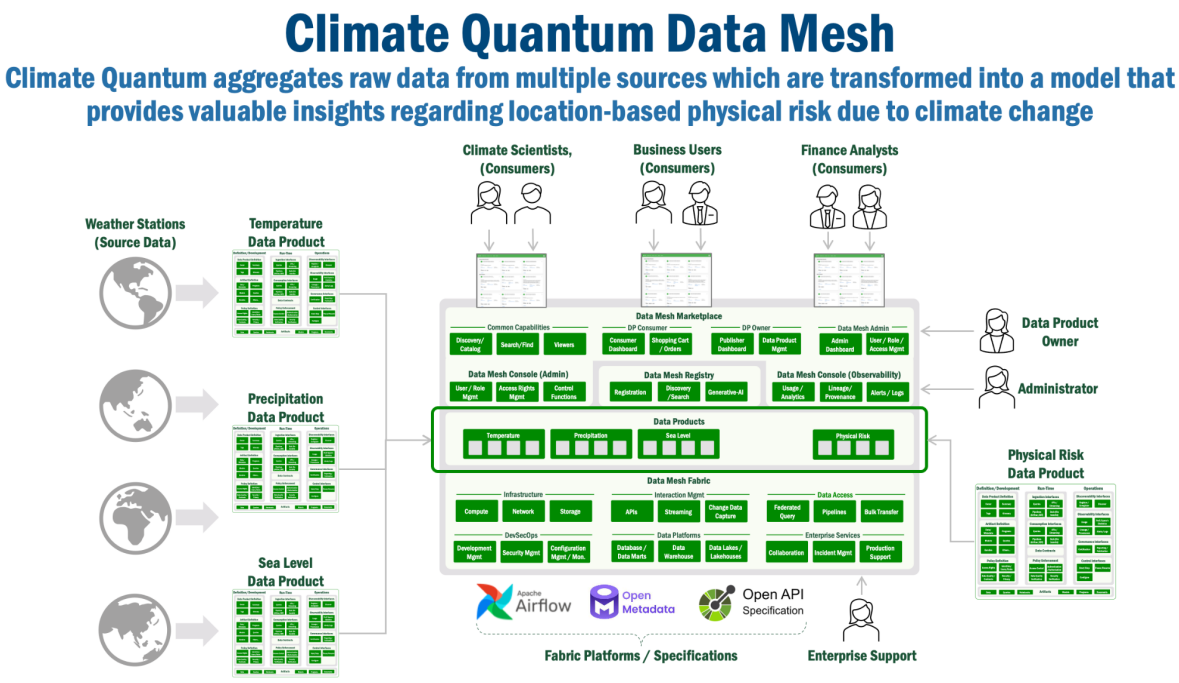


Figure 4-5. Climate Quantum Data Mesh

Climate Quantum's data mesh is a sophisticated and dynamic ecosystem, designed to meet the diverse needs of its users. From climate scientists to financial analysts, the data mesh provides accessible, reliable, and comprehensive climate data, empowering its users to make informed decisions in the face of climate change. Supported by a robust infrastructure and a team dedicated to its maintenance, Climate Quantum's data mesh stands as a model for collaborative, data-driven approaches to understanding and addressing global challenges.

Climate Quantum data mesh is an ecosystem of interconnected network of data products. There are several intermediate or "lower-level" data products that focus on key climate metrics:

temperature, precipitation, and sea level. These intermediate public data products form the backbone of Climate Quantum's data infrastructure, drawing in source information from a myriad of weather stations and transforming this raw data into a consistent and comprehensive set of climate data ensuring data accuracy and uniformity across the various inputs.

At the apex of Climate Quantum's data offerings is the primary public data product known as the Physical Risk model. This model integrates and synthesizes data from the temperature, precipitation, and sea level products to create a detailed analysis of the physical risks posed by climate change.

The consumers of these data products are as diverse as the data itself. Climate scientists delve into the data to predict and anticipate climate change trends, utilizing the detailed analyses to inform their research and studies. Business users leverage this data to communicate the impacts of climate change, translating complex climate metrics into actionable insights for organizational strategies. Financial analysts, particularly interested in the physical risk data product, use these insights to understand how climate change might affect the assets they manage, integrating this information into their financial models and risk assessments.

Each of these consumer groups relies on the Data Mesh Marketplace – a user-friendly interface that simplifies the process of finding, consuming, sharing, and trusting the climate data within the data mesh. The marketplace, bolstered by an integrated registry, offers a lightweight and low-maintenance solution for accessing the vast array of data products. This integration ensures that the marketplace remains up-to-date and reflective of the latest data offerings, facilitating easy access for all users.

While the marketplace is a primary interface for users, the Data Mesh Console and its command-line interface provide another layer of interaction. This tool is especially useful for data mesh administrators and data product owners, who often use it to execute specific commands and manage various aspects of the data products. The console's advanced capabilities make it a versatile tool for more technical users, enabling them to manage and interact with the data mesh in a more direct and granular way.

The architecture of Climate Quantum's data mesh is supported by a suite of platforms and specifications, integral to the fabric of the data mesh. For instance, pipelines are managed using Apache Airflow, a tool that streamlines and automates the data flow processes. Metadata management, an essential aspect of

maintaining data integrity and usability, is handled by OpenMetaData, ensuring that data lineage and documentation are kept accurate and up-to-date. And APIs in Climate Quantum's data mesh adhere to the OpenAPI specification, fostering standardization and interoperability across various data products and services. This adherence ensures that APIs are consistent, well-documented, and easy to use, facilitating seamless interaction between different components of the data mesh.

These platforms and tools are supported by the enterprise support team, a group dedicated to maintaining the underlying infrastructure and common platforms that support the data mesh. Their role is crucial in ensuring that the data mesh operates smoothly, efficiently, and without interruption..

Within this ecosystem, each data product, while consuming the infrastructure, interaction, data access, DevSecOps, and data management services provided by the enterprise, retains the autonomy to choose how best to utilize these services. This flexibility is crucial, as it allows data product owners to tailor their use of enterprise services according to the specific needs and requirements of their data products and their consumers.

Chapter 5. Driving Data Products with Data Contracts

A NOTE FOR EARLY RELEASE READERS

With Early Release ebooks, you get books in their earliest form—the author's raw and unedited content as they write—so you can take advantage of these technologies long before the official release of these titles.

This will be the 5th chapter of the final book. Please note that the GitHub repo will be made active later on.

If you have comments about how we might improve the content and/or examples in this book, or if you notice missing material within this chapter, please reach out to the editor at sevans@gmail.com.

In this chapter, we will first start by looking at what Data Mesh is from an implementation perspective, answering the question, what are its main components? I will then draw the parallel with product thinking, explore what a data product is, and finally jump into data contracts. The examples in this chapter follow the theme of Climate Quantum Inc.

Bringing Value Through Trust

Do not worry, I am not switching from an engineering-oriented book to a business book, but I am convinced that, collectively, we need to keep goals with Data Mesh, and one of them is trust.

Rooted in agile methodologies, Data Mesh focuses on bringing value to the enterprise. I know that, from an engineering and technology perspective, it seems weird to talk about value. After all, what is “value” in data engineering?

When I am thinking about a good product, I think a lot about the trust I have in this product. When I have the time, I love to take photos and use my rather nice Nikon mirrorless camera. As I hike and get my camera out, I expect it to be available as soon as I turn it on, not having some weird boot time. When I press the actuator, I expect the photo to be available instantly on my CFexpress card and within ten seconds on my iPhone.

The quality here is that the image is saved on the card, matching my expectations in resolution, compression or not, color balance, and a few more attributes. The time needed to transfer my photo from the camera to my phone is a service-level objective.

In many conversations with fellow engineers and scientists, I often tell them that I am not smart enough to know what to do with data, but I know how to bring them the data they need (and sometimes want). And my teams deliver either the data or the tools to access & process the data.

I don't pretend to know my customers' jobs, although I spend time learning about them, and everybody should. I try to nurture an honest and **positive relationship** with them. It can be challenging, as you probably experienced in your career.

The second element of trust is showing **expertise**: of course, I can pretend I am an expert, but what can describe this expertise in a strong way: a contract. The data contract, see p. TBD in this chapter, will provide the required level of expertise needed to trust the data.

Consistency is the third element of trust described by Harvard Business Review (https://hbr.org/2019/02/the-3-elements-of-trust). Once more, the data contract will provide this consistency through data quality results and service-level indicators (SLI).

As a result, my customers get access to data they can trust and build upon. Trust is the value you should aim to deliver.

Am I bringing business considerations into a technical book? Everything you will be doing from now on is going to be based on the trust you are going to build, and you are going to guarantee this trust with a series of contracts, as illustrated by Figure 5-1.

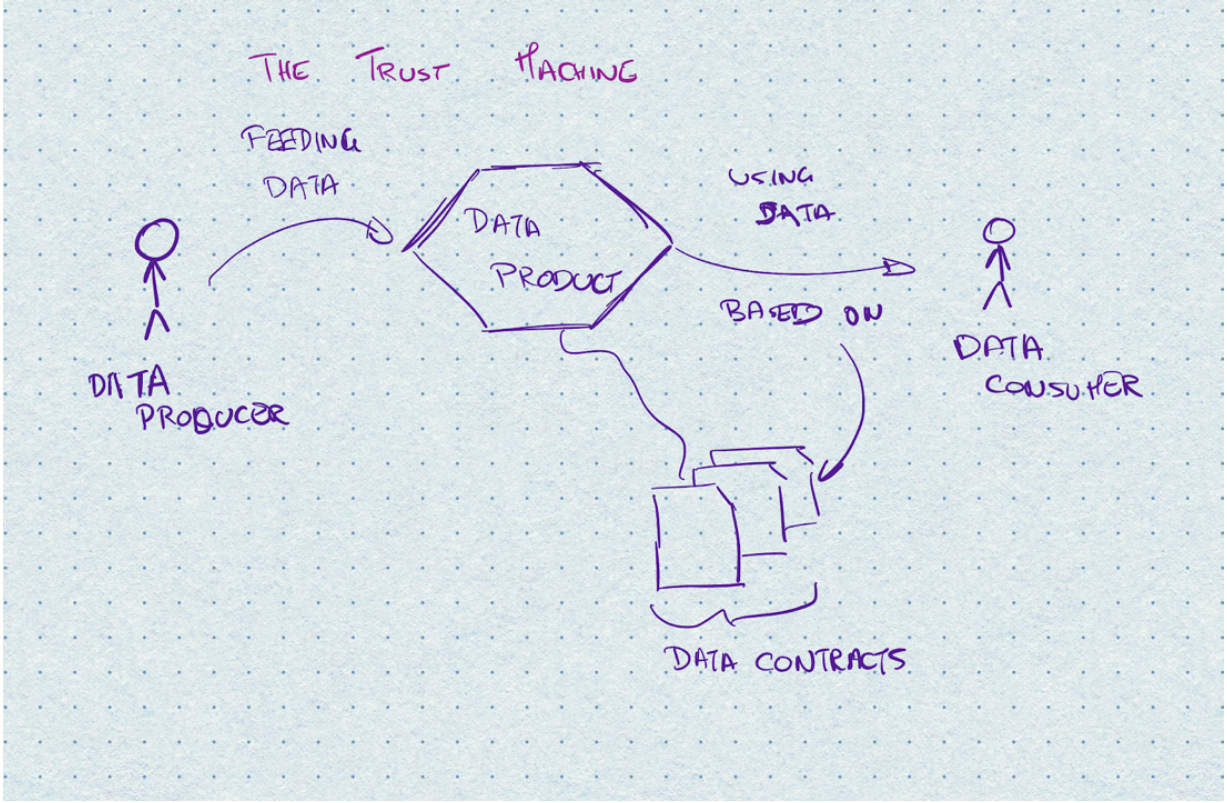


Figure 5-1. : Assuming your goal is to provide trusted data to your consumer, the data contracts are going to be the vehicle of this trust.

Now that I have established the need for trust as the main value, let's dive into product thinking before we can articulate this trust and focus on implementing it.

Navigating the data contract

As you saw in the first part of this chapter, the need for a data contract is pretty obvious: you need an artifact to support the trust you want to build in your data, and as you read in Chapter 2, many of the characteristics and features of the data products. Let's go first into the theory, understanding the information you want in the contract, and dive into a couple of examples.

Going through the theory

Let's start with the boring part: the theory. A contract establishes a formal relationship between two or more parties. We have all signed implicitly or explicitly contracts: you're working for someone (or you have customers), you most likely have a cell phone, and you probably don't live under a bridge. As you most likely imagined, a data contract is a similar agreement that now involves data (duh).

A data contract acts as an agreement between multiple parties, specifically, a producer and consumer(s).

I like to compare a data contract to buying a car (clarification needed: all data producers are not used-car sales guys). Imagine I want a 2013 Volkswagen Beetle 2.0L TDI. I like the shape of the

Beetle; I am a bit flower power, I believe (at times) in Diesel, and a 2.0L engine for 3117 pounds / 1.4T should give me a nicely reactive car. Those features are part of the contract.

In a data contract, those features can be the schema. My dataset contains a customer table with twenty-five columns, one column being the first name, which is of type VARCHAR, and a length of 32.

The contract includes service-level agreements (SLA) and functional/non-functional requirements (NFR).

For my little VW Beetle, it could be the 32 MPG/7.4 L/100Km consumption or when the garage will have it available so I can pick it up. For data, it can be time to detect an issue or, for a daily batch, the time the data is available.

In some cases, problems can be a total deal breaker: too many duplicated records, too many NULLs in a required field, or heavier nitrogen-oxide emissions (NOx) released in the atmosphere. The constraints can come from the producer/seller or the regulator -- and this opens the door to federated computational governance, but I won't go there in this chapter, but you can check in TBD.

Stacking up good information

There are many flavors of data contracts; in this book, I will actively follow the Open Data Contract Standard (ODCS), which is part of the Bitol project supported by the Linux Foundation AI & Data (see https://bitol.io).

So, what do you put in a data contract? I'd say virtually anything. In the open source data contract standard (see https://aidaug.org/odcs), the community has divided the information into the following eight categories:

Let's go through each category and detail it.

Fundamentals & demographics

This section contains general information about the contract, like name, domain, version, and much room for information.

Dataset & schema

This section describes the dataset and the schema of the data contract. It is the support for data quality, which I detail in the next section. A data contract focuses on a

single dataset with several tables (and, obviously, columns).

Data quality

This category describes data quality rules & parameters. They are tightly linked to the schema defined in the dataset & schema section.

Pricing

This section covers pricing when you bill your customer for using this data product. Pricing is currently experimental.

Stakeholders

This important part lists stakeholders and the history of their relation with this data contract.

Roles

This section lists the roles that a consumer may need to access the dataset depending on the type of access they require.

Service-level agreement

This section describes the service-level agreements (SLA).

Custom properties

> This section covers custom & other properties in a data contract using a list of key/value pairs. This section gives flexibility without creating a new template version whenever someone needs additional properties.

The data contract itself uses a YAML file. The choice of YAML was evident as it can be read equally easily by machines & humans. YAML files can live extremely well in GitHub repositories, where source control provides excellent traceability.

Let's see how data contracts relate to data products and Data Mesh. It will help you understand how you can optimally use them.

It's all about proper versioning

Like my car example, data can be sold (or consumed) as a product. Like my Beetle, it can have functional requirements (four wheels, a steering wheel, and a few other details), non-functional requirements (the quantity of NOx released in the atmosphere while driving), and SLA (delivery date, replacement car when stranded).

However, when I see the NOx problems, I may switch to a non-diesel version of the Beetle, like the 2014 2.5L five-cylinder. It is still the same product, not the same version (2014 vs. 2013), and a variation on the product (gas vs. diesel).

For a data product, it's the same thing. I can add or remove some information and deal with the changes through versions. I can also have different variations on the product I share, like raw, curated, aggregated, and more.

To summarize, a data product can have multiple data contracts, exactly one data contract per pair of datasets/versions. It gives us the required flexibility.

Keeping it simple and semantic

Changes follow semantic versioning to guarantee consistency in how data engineers version data contracts. Semantic versioning relies on patch, minor, and major changes. Let's look at a few examples in the *Documenting in a slightly better way* section.

Data Mesh tracks data product owners (DPO) as stakeholders in any data product. When they move to a new role, we add the new DPO to the contract, keeping a human lineage. This results

in a patch version: my data contract can move from 1.0.1 to 1.0.2.

If I add an age column to my prospect table, part of my curated customer data set in my customer data product. The data contract is bumping to a minor version for this dataset, from version 1.0.2 to 1.1.0.

Over time, the data producer realized that the prospect and customer tables should be combined. It is a major change, breaking much of the consumer code. This evolution is a reason to bump the updated contract to version 2.0.0.

A *major change* is a change that does not provide backward compatibility. Such a change will cause existing downstream apps/solutions/queries to break.

A *minor change* maintains backward compatibility; it allows existing downstream apps/solutions/queries to function with no issue.

A *patch* is a bug fix that provides backward compatibility. It can also be an information change like a new stakeholder in a data contract.

Table 5-1 lists changes considered patches, minor, and major with examples. These lists are not exhaustive.

SEVERITY OF CHANGES

Changes are not equal in intensity. Table 5-1 describes the severity of the change, whether it is a patch, a minor, or a major change for tables, APIs, and data contracts.

Table 5-1. : Severity of changes in your data product

Artifact	Patch	Minor	Major
Table		Adding a new column. Logic change to an existing column.	Column data type change. Column name change. Removal of a column.
API	A bug fix that is not a minor or major change.	Adding a new field in the payload. Adding an optional parameter in the HTTP request. Changing a required parameter to optional.	Changing the format of request or response data. Changing the data type of a resource (like changing from a string to an integer). Removing one or more resources, removing or changing

Artifact	Patch	Minor	Major
			properties or methods for a particular resource, or any other changes to API functionality. Adding a new required field for client HTTP requests.

Artifact	Patch	Minor	Major
Data contract	Updates to metadata. Changes to description fields. Changes to stakeholders.	Revised data quality rule. Adding a new key, given that it is optional or required with a default value. Adding a custom property.	Changing the data type of a key. Changing the name of a key. Removing a key.

It may not sound easy, but it will allow data producers and consumers to evolve at their own pace, increasing trust between them and the robustness of our systems.

To ease the management of those contracts, Data Mesh needs tooling. For example, the Rosewall team developed a comparison service to analyze two contracts and suggest a version number.

A philosophy change may also be needed, as a strict contract does not mean that its consumption should be as uncompromising. After all, Jon Postel, one of the creators of the TCP protocol, said, “Be conservative in what you do, be liberal in what you accept from others.” It became the robustness principle in computer science and is often called Postel’s law.

As this chapter focuses on data products and data contracts, I ignore the role of Data Mesh here. Still, you can easily picture that Data Mesh is a way to organize data products, as illustrated in Figure 5-2.

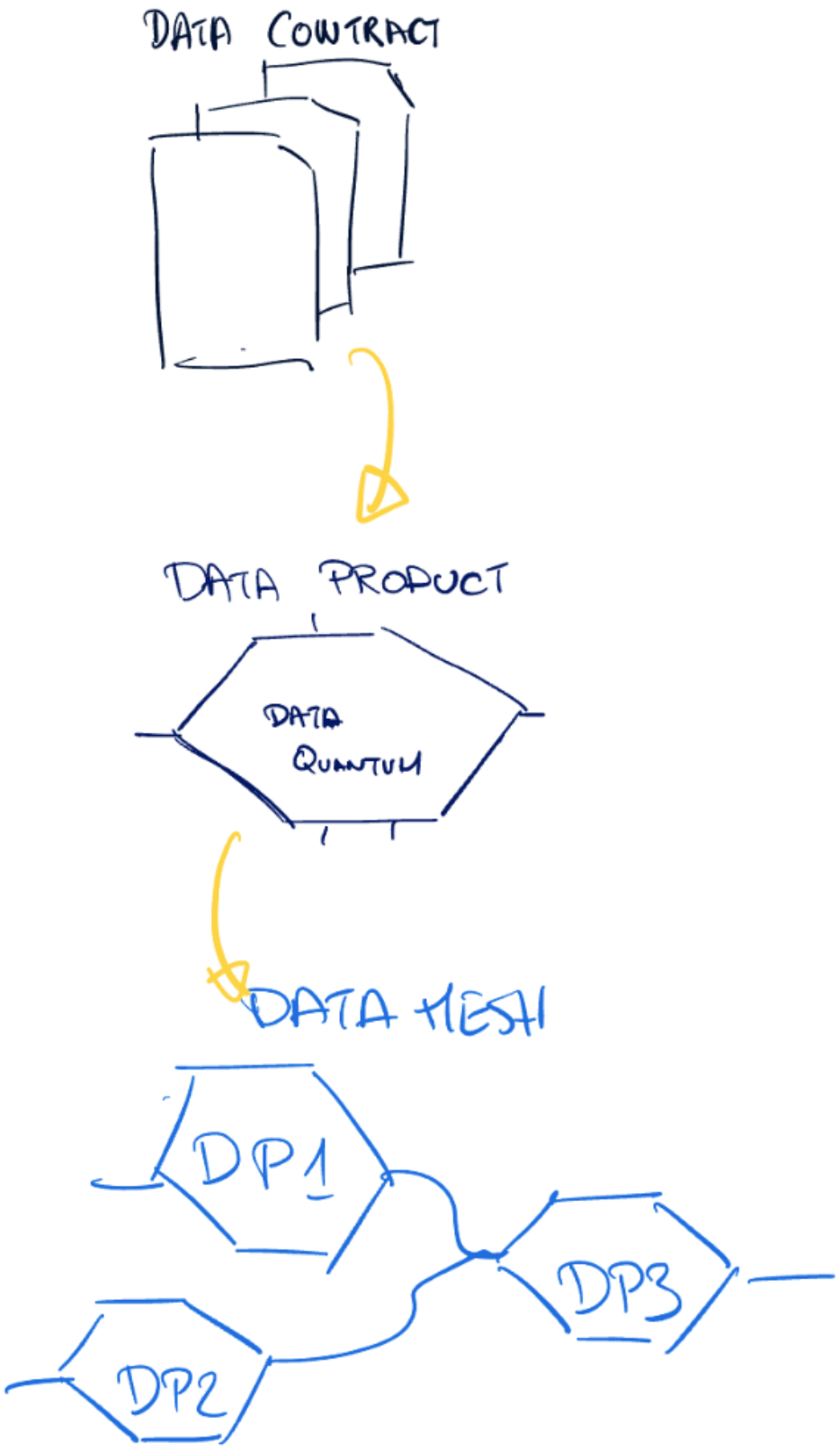

DATA CONTRACT
DATA PRODUCT
DATA QUANTUM
DATA MESH
DP1
DP2
DP3

Figure 5-2. : The hierarchy between the different artifacts.

Walking through an example: complementing tribal knowledge

Let's dive into a complex business problem that the data contract solves. There are a few examples in the GitHub repository at https://github.com/AIDAUserGroup/data-contract-template, and the team will continue adding more examples.

However, I wanted to document a problem we can solve with data contracts: tribal knowledge. Tribal knowledge happens when a group of people knows something, and the information is relatively hard to get to when you're not part of this group. It's an oral tradition. Tribal knowledge is not wrong per se, but it's hard to scale, does not resist organizational changes, and will create issues in a regulatory environment. One should not get rid of it but learn how to live with it and complement it. Here are two ways of enhancing it:

- Document, document, document.
- Create a human lineage.

Documenting in a slightly better way

Usually, engineers do not mind documenting. They are often under or over-documenting things, but documentation usually exists in the 21st century. The problem is documenting change. The data contract provides a solution to documenting at the right level and synchronizing change.

Here, the data contract can help. For example, leverage the description and other informational fields. Note: this data contract format focuses for now on table and structured data, extensions are in progress.

```
    - table: tbl
      description: Provides core payment metrics
      dataGranularity: Aggregation on columns txn_r
      columns:
        - column: txn_ref_dt
          businessName: Transaction reference date
          logicalType: date
          physicalType: date
          description: Reference date for the trans
          sampleValues:
            - 2022-10-03
            - 2025-01-28
```

But this is an easy example, although sometimes, finding the right field is not as easy as it seems!

Sometimes, fields result from a calculation or have business rules associated with them. The data contract lets you materialize those constraints via authoritative definitions. The following example shows that the rcvr_cntry_code field has both a business definition and a reference implementation.

```
    - table: tbl
      columns:
      - column: rcvr_cntry_code
        businessName: Receiver country code
        logicalType: string
        physicalType: varchar(2)
        authoritativeDefinitions:
          - url: https://collibra.com/asset/748f-71
            type: Business definition
          - url: https://github.com/myorg/myrepo
            type: Reference implementation
```

Now, it is up to us to define the policies we want to enforce: % of completion of documentation, % of fields linked to authoritative definitions, and more. No more under or over-documentation, no more out-of-sync between your data model and your documentation. The data contract provides a rich single source of truth.

Creating a human lineage

Let’s understand what a human lineage (instead of a data lineage) can do to share knowledge and track the history. As the story evolves, I will translate it into code used in the data contract. Data lineage applies to following the journey of data.

Let’s imagine this scenario: Clint Eastwood joined Climate Quantum, Inc. a few years ago as a data product owner (DPO).

```
stakeholders:
  - username: ceastwood
    role: dpo
    dateIn: 2014-08-02
```

Quickly, thanks to his intolerant, direct, and fiery attitude, he went exploring other areas and was replaced by John Wayne.

```
stakeholders:
  - username: ceastwood
    role: dpo
    dateIn: 2014-08-02
    dateOut: 2014-10-01
    replacedByUsername: jwayne
  - username: jwayne
    role: dpo
    dateIn: 2014-10-01
```

John Wayne’s style was a better fit, and he got promoted. He hired Calamity Jane to replace him as a DPO.

```
stakeholders:
  - username: ceastwood
    role: dpo
    dateIn: 2014-08-02
    dateOut: 2014-10-01
    replacedByUsername: jwayne
  - username: jwayne
    role: dpo
    dateIn: 2014-10-01
    dateOut: 2019-03-14
    replacedByUsername: cjane
  - username: cjane
    role: dpo
    dateIn: 2019-03-14
```

Great things happened, and Calamity got to her sabbatical. Although she was still going to be the DPO when she returned, she asked a kid, Billy, to watch for her.

```
stakeholders:
  - username: ceastwood
    role: dpo
    dateIn: 2014-08-02
    dateOut: 2014-10-01
    replacedByUsername: jwayne
  - username: jwayne
    role: dpo
    dateIn: 2014-10-01
    dateOut: 2019-03-14
    replacedByUsername: cjane
  - username: cjane
    role: dpo
```

```
      dateIn: 2019-03-14
      comment: Minor interruption due to sabbatical
      dateOut: 2021-04-01
      replacedByUsername: bkid
    - username: bkid
      role: dpo
      dateIn: 2021-04-01
```

And that's when Billy got into trouble and needed an emergency leave. As I jinxed Murphy's law too many times, this is when I needed critical information from the DPO. Although Billy and Calamity were out, thanks to the human lineage described in the contract, I could happily reach out to Clint (yes, John was also off this week).

This example illustrates a fictional situation, but I am pretty sure you have experienced something similar in your career.

What is Data QoS, and why is it critical?

Normalizing the way we describe data quality and service levels is key to the success of your data contract. In this section, I am introducing the notion of data quality of service (Data QoS), which is the result of combining Data Quality (DQ) with Service-Level Agreements (SLA). I will start by explaining the concept, and I will then drill down to describe the elements

composing the Data QoS, focusing first on Data Quality and then on Service-Level Indicators. Finally, I will explain how I grouped them.

Quality of Service (QoS) is a well-established concept in network engineering. QoS is the measurement of the overall performance of a service, such as a telephony, computer network, or cloud computing service, particularly the performance seen by the network users. In networking, several criteria are considered to quantitatively measure the QoS, such as packet loss, bit rate, throughput, transmission delay, availability, and more. This chapter applies QoS to data engineering.

Data Quality of Service (Data QoS)

As your need for observing your data grows with the maturity of your business, you will realize that the number of attributes you want to measure will bring more complexity than simplicity. That's why, back in 2021, I came up with the idea to combine both into a single table, inspired by Mandeleev's (and many others) work on classifying atomic elements in physics, you can see the result in figure 5-3.

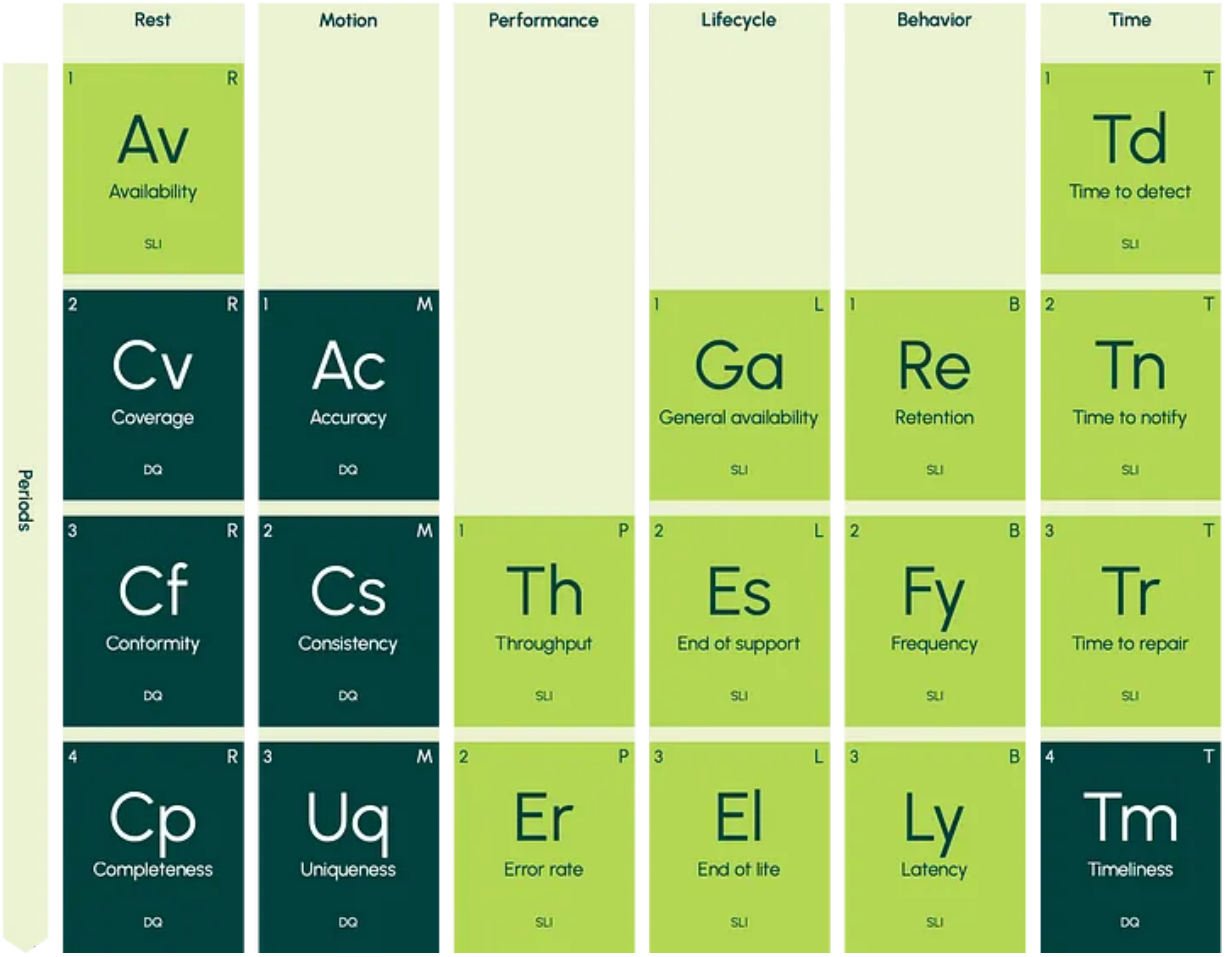


Figure 5-3. : Inspired by Mandeleev's periodic table for classifying atomic elements, the Data QoS table represents the finest elements used for measuring data quality and service levels for data.

Let's have a look at data quality first.

Representation

To represent the elements, I needed to identify precisely each element on two axes:

1. Time (or period).

2. Group.

Each element received additional attributes, as shown in figure 5-4.

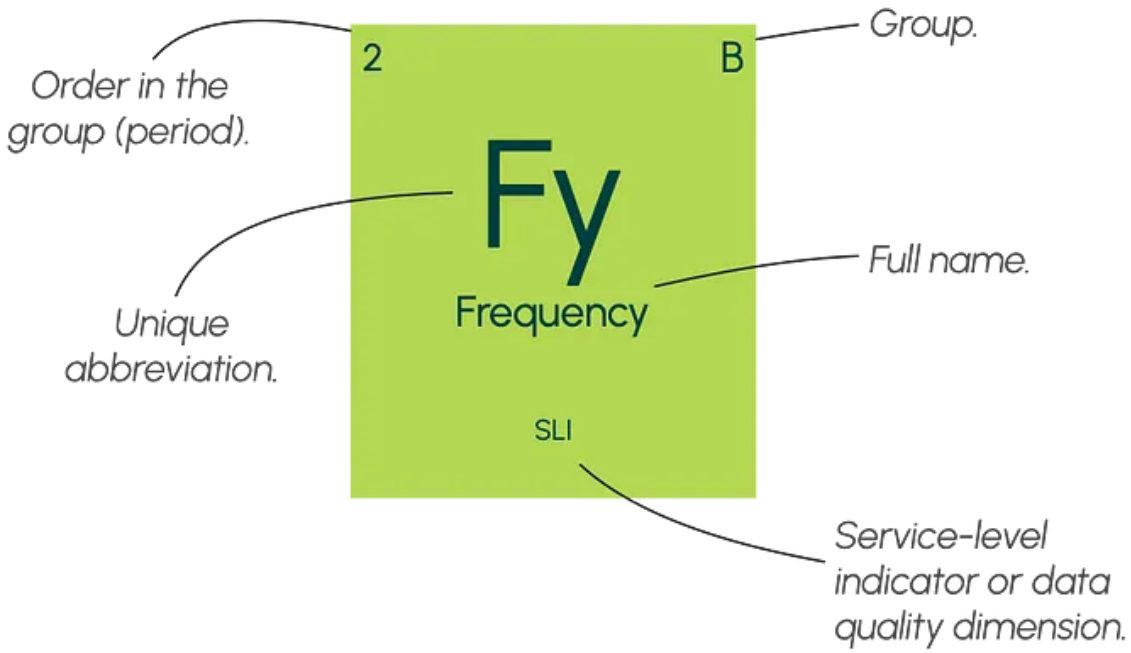


Figure 5-4. : Each element has a name, an abbreviation, a group, an order in this group, and a category.

Periods & time-related

The periods are time-sensitive elements. Some elements are pretty obvious, as “end of life” is definitely after “general availability,” as illustrated by Figure 5-7.

Figure 5-5. : General availability comes before the end of support, which comes before the end of life.

Classification of some elements is more subtle: when data comes to your new data store, you will check accuracy before consistency, and you can check uniqueness only when you have significant data. The elements have no chronological link, but they happen in sequence, as illustrated by Figure 5-6.

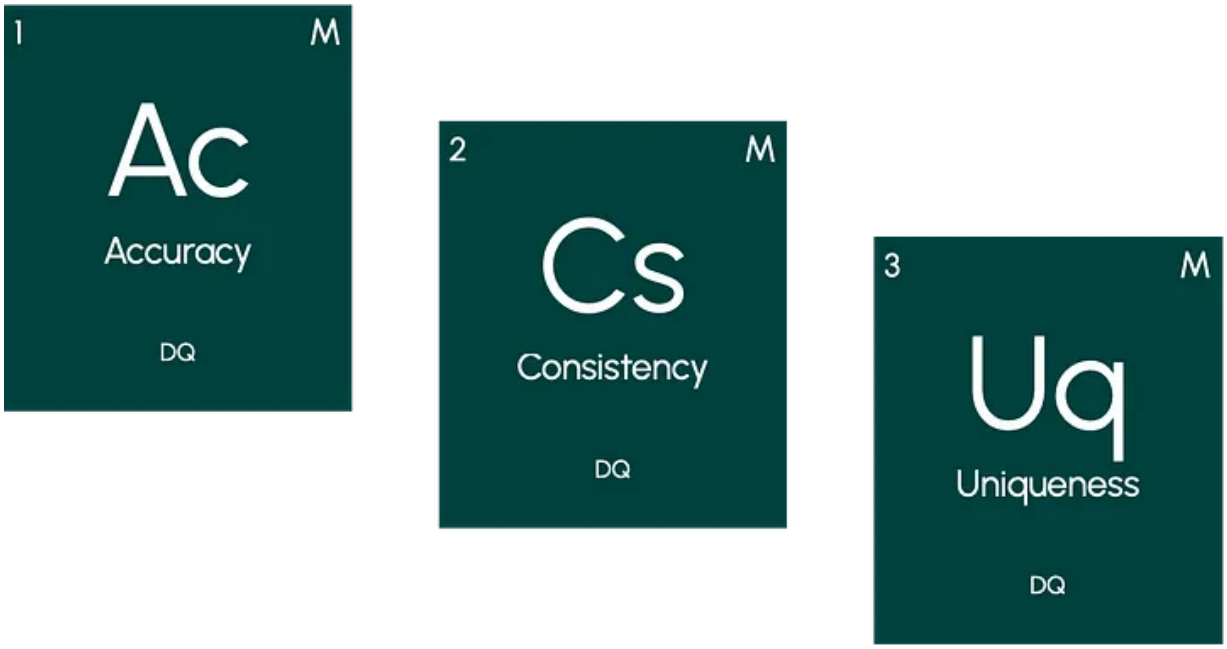


Figure 5-6. : Checking accuracy, consistency, and uniqueness happens sequentially.

Grouping

The second classification to find was about grouping. How can we group those elements? Is there a logical relation between the elements that would make sense?

This is what I came up with:

- Data at rest (R).
- Data in motion (M).
- Performance (P).
- Lifecycle (C) of the product itself.
- Behavior (B) of the data includes retention, refresh frequency, availability time, and latency.
- Time (T)-related indicators.

Why does it matter?

There are a lot of benefits to the classification and definition of the elements forming the Data QoS, as in the service-level indicators and the data quality dimensions.

> *Definitions we can agree on*
>
> > The first step of the Information Technology Infrastructure Library (ITIL) is to set up a common

vocabulary among the stakeholders of a project. Although ITIL might not be adequate for everything, this first step is crucial. Data QoS offers an evolutive framework with consistent terms and definitions.

Compatibility with data contracts

As we are focusing on data contracts, you need to keep in mind that data contracts need to be built on standardized expectations. It's obvious for the data retention period, as you would probably not see duration, safekeeping, or something else. However, latency and freshness are often interchanged; let's go for latency.

Setting the foundation

Data QoS is not carved in stone, even if it could be compared to the Rosetta stone. It supports evolution and innovation while delivering a solid base.

Data Quality is not enough

Regarding data, the industry standard for trust has often been limited to data quality.

I felt this for a long time. In 2017, at Spark Summit (https://youtu.be/ka8xhQAoj-E?t=265), I introduced Cactar (Consistency, Accuracy, Completeness, Timeliness, Accessibility, and Reliability) as an acronym for six data quality dimensions relayed in a Medium article (https://medium.com/%40jgperrin/meet-cactar-the-mongolian-warlord-of-data-quality-d7bdbd6a5398). Although there is no official standard, the EDM Council (https://www.edmcportal.org/glossary/data-quality-dimensions/) had a few different ones and added a 7th one. So I decided to align on the EDM Council's list.

Here are the seven data quality dimensions, as represented by figure 5-7.

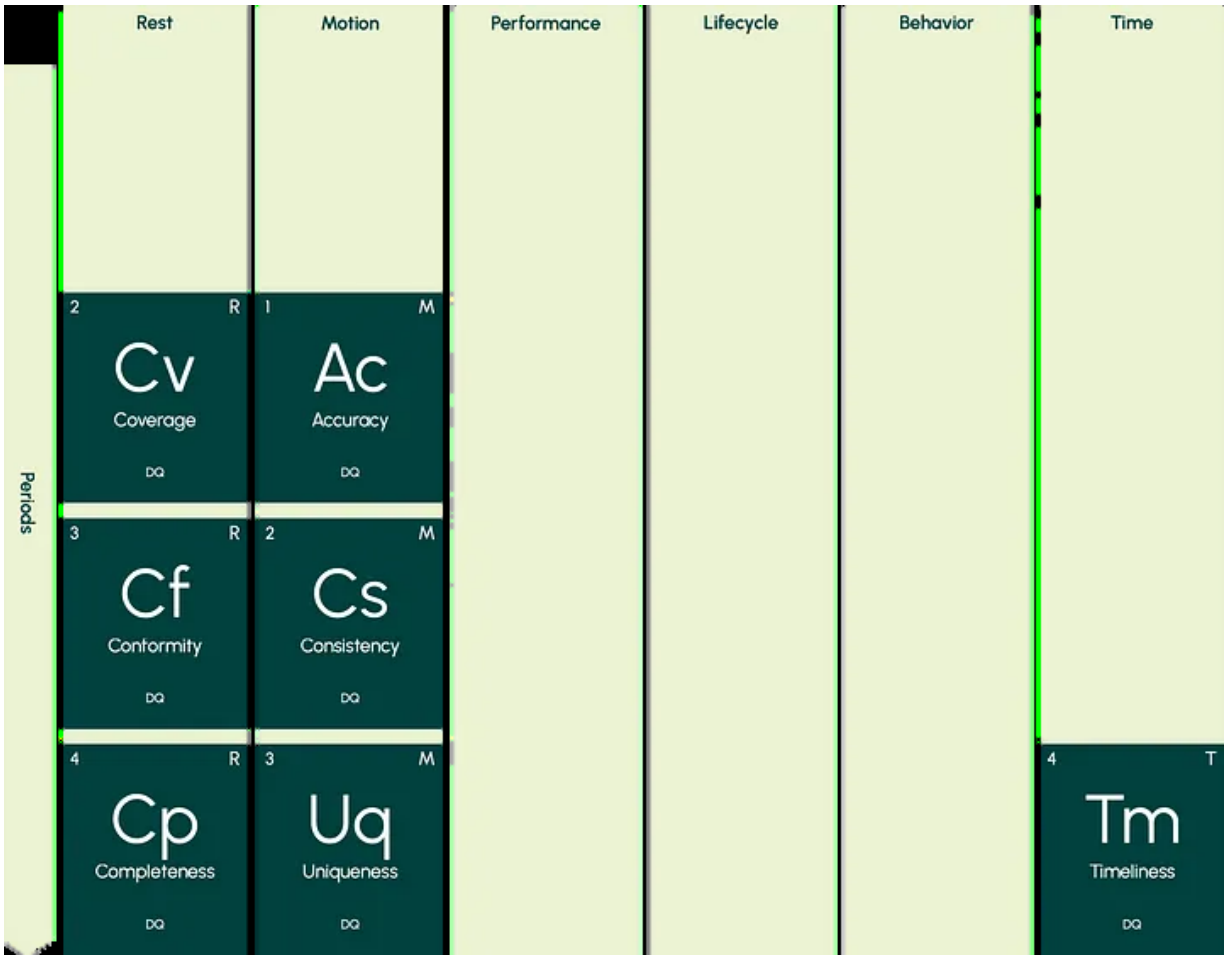


Figure 5-7. : The seven data quality dimensions are on the Data QoS table.

Accuracy (Ac)

The measurement of the veracity of data to its authoritative source: the data is provided but incorrect. Accuracy refers to how precise data is, and it can be assessed by comparing it to the original documents and trusted sources or confirming it against business rules.

Examples:

- A customer is 24 years old, but the system identifies them as 42 years old.
- A supplier address is valid, but it is not their address.
- Fractional quantities are rounded up or down.

Fun fact: a lot of accuracy problems come from the data input. If you have data entry people on your team, reward them for accuracy, not only speed!

Completeness (Cp)

Data is required to be populated with a value (aka not null, not nullable). Completeness checks if all necessary data attributes are present in the dataset.

Examples:

- A missing invoice number when it is required by business rules or law.
- A record with missing attributes.
- A missing expiration month in a credit card number.

Fun fact: a primary key is always a required field.

Conformity (Cf)

Data content must align with required standards, syntax (format, type, range), or permissible domain values. Conformity assesses how closely data adheres to standards, whether internal, external, or industry-wide.

Examples:

- The customer identifier must be five characters long.
- The customer address type must be in the list of governed address types.
- Merchant address is filled with text but not an identifying address (invalid state/province, postal codes, country, etc.).
- Invalid ISO country codes.

Fun fact: ISO country codes are 2 or 3 digits (like FR and FRA for France). If you mix up the two in the same datasets, it's not a conformity problem; it's a consistency problem.

Consistency (Cs)

Data should retain consistent content across data stores. Consistency ensures that data values, formats, and definitions in one group match those in another group.

Examples:

- Numeric formats converted to characters in a dump.
- Within the same feed, some records have invalid data formats.
- Revenues are calculated differently in different data stores.
- String are shortened from a max length of 255 to 32 when they go from the website to the warehouse system.

Fun fact: I was born in France on 05/10/1971, but I am a Libra (October). When expressed as strings, date formats are transformed through a localization filter. So, being born on October 5th makes my date representation 05/10/1971 in Europe, but 10/05/1971 in the U.S

Coverage (Cv)

All records are contained in a data store or data source. Coverage relates to the extent and availability of data present but absent from a dataset.

Examples:

1. Every customer must be stored in the Customer database.
2. The replicated database has missing rows or columns from the source.

Timeliness (Tm)

The data must represent current conditions; the data is available and can be used when needed. Timeliness gauges how well data reflects current market/business conditions and its availability when needed

Examples:

- A file delivered too late or a source table not fully updated for a business process or operation.
- A credit rating change was not updated on the day it was issued.
- An address is not up to date for a physical mailing.

Fun fact: Forty-five million Americans change addresses every year.

Uniqueness (Uq)

How much data can be duplicated? It supports the idea that no record or attribute is recorded more than once. Uniqueness means each record and attribute should be one-of-a-kind, aiming for a single, unique data entry (yeah, one can dream, right?)

Examples:

- Two instances of the same customer, product, or partner with different identifiers or spelling.
- A share is represented as equity and debt in the same database.

Fun fact: data replication is not bad per se; involuntary data replication is!

Let's agree that those seven dimensions are pretty well-rounded. As an industry, it's probably time to say: good enough. Of course, it completely ruins my Cactar acronym (and its great backstory).

But I still feel it is not enough. Data quality does not answer questions about end-of-life, retention period, and time to repair when broken. Let's look at service levels.

Service-levels complement quality

As much as data quality describes the condition of the data, service levels will give you precious information on the expectations around availability, the condition, and more.

The quality here is that the image is saved on the card, matching my expectations in resolution, compression or not, color balance, and a few more attributes. The time needed to transfer my photo from the camera to my phone is a service-level objective.

Figure 5-8 is a list of service-level indicators that can be applied to your data and its delivery. You will have to set some objectives (service-level objectives or SLO) for your production systems and agree with your users and their expectations (set service-level agreements or SLA).

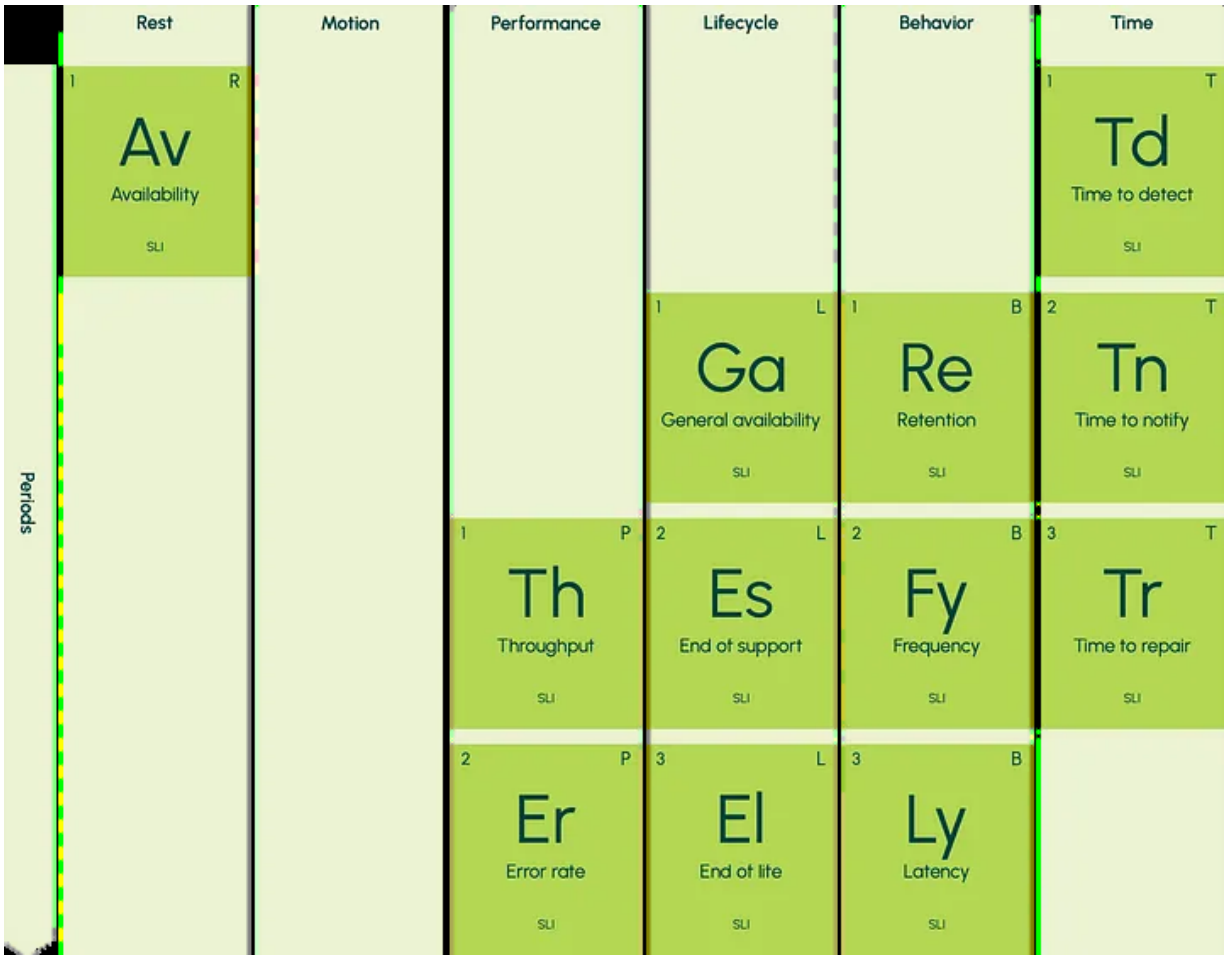


Figure 5-8. : The service levels on the Data QoS table.

Availability (Av)

In simple terms, the question is: Is my database accessible? A data source may become inaccessible for various reasons, such as server issues or network interruptions. The fundamental requirement is for the database to respond affirmatively when you use the JDBC's connect() method.

Throughput (Th)

Throughput is about how fast I can access the data. It can measured in bytes or records by unit of time.

Error rate (Er)

How often will your data have errors, and over what period? What is your tolerance for those errors?

General availability (Ga)

In software and product management, general availability means the product is now ready for public use, fully functional, stable, and supported. Here, it applies to when the data will be available for consumption. If your consumers require it, it can be a date associated with a specific version (alpha, beta, v1.0.0, v1.2.0…).

End of support (Es)

The date at which your product will not have support anymore.

For data, it means that the data may still be available after this date, but if you have an issue with it, you won't be offered a fix. It also means that you, as a consumer, will expect a replacement version.

Fun fact: Windows 10 is supported until October 14, 2025.

End of life (El)

The date at which your product will not be available anymore. No support, no access. Rien. Nothing. Nada. Nichts.

For data, this means that the connection will fail or the file will not be available. It can also be that the contract with an external data provider has ended.

Fun fact: Google Plus was shut down in April 2019. You can’t access anything from Google’s social network after this date.

Retention (Re)

How long are we keeping the records and documents? There is nothing extraordinary here, as with most service-level indicators, it can vary by use case and legal constraints.

Frequency of update (Fy)

How often is your data updated? Daily? Weekly? Monthly? A linked indicator to this frequency is the time of availability, which applies well to daily batch updates.

Latency (Ly)

Measures the time between the production of the data and its availability for consumption.

Time to detect (an issue) (Td)

How fast can you detect a problem? Sometimes, a problem can be breaking, like your car not starting on a cold morning or slow, like data feeding your SEC (Security Commission for Publicly Traded Companies) being wrong for several months. How fast do you guarantee the detection of the problem? You can also see this service-level indicator called “failure detection time.”

Fun fact: squirrels (or another similar creature) ate the gas line on my wife’s car. We detected the problem as the gauge went down quickly, even for a few miles. Do you even drive the car to the mechanic?

Time to notify (Tn)

Once you see a problem, how much time do you need to notify your users? This is, of course, assuming you know your users.

Time to repair (Tr)

How long do you need to fix the issue once it is detected? This is a very common metric for network operators running backbone-level fiber networks

Of course, there are a lot more service-level indicators that will come over time. Agreements follow indicators; agreements can include penalties. You see that the description of the service can become very complex.

In the next section, let's apply Data QoS to the data contract, in the context of Climate Quantum, Inc.

Applying Data QoS to the data contract

In this section, let's look at a three examples on how we apply Data QoS, the combination of data quality and service levels, to the data contract.

To demonstrate the use of those dimensions in a data contract, I will use the NYC Air Quality dataset, which could be used by Climate Quantum, Inc. The dataset's metadata looks like table 5-2.

Table 5-2. : Name, description, and type of the columns of the NYC Air Quality dataset.

Column Name	Description	Type
UniqueID	Unique record identifier	Plain Text
IndicatorID	Identifier of the type of measured value across time and space	Number
Name	Name of the indicator	Plain Text
Measure	How the indicator is measured	Plain Text
MeasureInfo	Information (such as units) about the measure	Plain Text
GeoTypeName	Geography type; UHF’ stands for United Hospital Fund neighborhoods; For instance, Citywide, Borough, and Community Districts are different geography types	Plain Text

Column Name	Description	Type
GeoJoinID	Identifier of the neighborhood geographic area, used for joining to mapping geography files to make thematic maps	Plain Text
GeoPlaceName	Neighborhood name	Plain Text
TimePeriod	Description of the time that the data applies to ; Could be a year, range of years, or season for example	Plain Text
StartDate	Date value for the start of the TimePeriod; Always a date value; could be useful for plotting a time series	Date & Time
DataValue	The actual data value for this indicator, measure, place, and time	Number

Column Name	Description	Type
Message	notes that apply to the data value; For example, if an estimate is based on small numbers we will detail here	Plain Text

Checking conformity of measurements

Let’s make sure that the information around measurements conforms to our expectations.

As a reminder, here are some conformity examples:

- The customer identifier should be eight characters long, and it is not.
- The customer address type must be in the list of governed address types (home, office).
- The address is filled with text but not an identifying address.
- Invalid ISO currency codes.
- Temperature contains letters.

In the data contract, this is how you could add a conformity data quality rule. In this scenario, the measurement identifier

requires a minimum value of 100,000. If not, the error is considered having an operational business impact.

```
  - table: Air_Quality
    description: Air quality of the city of New Yo
    dataGranularity: Raw records
    columns:
    - column: UniqueID
      isPrimary: true
      businessName: Unique identifier
      logicalType: number
      physicalType: int
          quality:
          - templateName: RangeCheck
            toolName: ClimateQuantumDataQualityPac
            description: 'This column should not c
            dimension: conformity
            severity: error
            businessImpact: operational
            customProperties:
              - property: min
                value: 100000
```

Completeness

Data is required to be populated with a value: you don’t want NULL values.

Here are some examples:

- A missing customer identifier, phone, or more.
- When null values are not allowed (required fields).
- A field must be populated per business rules.
- A record with missing attributes.

This is how you can represent completeness in a data contract: UniqueID is a required field. If this rule is not valid, it is an error with an operational business impact.

```
- table: Air_Quality
  description: Air quality of the city of New Yo
  columns:
  - column: UniqueID
    isPrimary: true
    businessName: Unique identifier
    logicalType: number
    physicalType: int
    quality:
    - templateName: NullCheck
      toolName: ClimateQuantumDataQualityPackage
      description: 'This column should not conta
      dimension: completeness
      severity: error
      businessImpact: operational
```

Accuracy

Insures that the provided data is correct. Here are a couple of examples:

- A customer is 12 years old, but the system identifies them as 32 years old.
- A supplier address is valid, but it is not the actual supplier's address.

In the data contract, you can specify that the value of the air quality should be between 0 and 500. As you can see, it is the same application (templateName) used for the validity dimension. The same tool can be used for multiple data quality dimensions.

```
- table: Air_Quality
  description: Air quality of the city of New Yo
  dataGranularity: Raw records
  columns:
  - column: DataValue
    businessName: Measured value
    logicalType: number
    physicalType: float(3,2)
        quality:
        - templateName: RangeCheck
          toolName: ClimateQuantumDataQualityPac
          description: 'This column should conta
          dimension: accuracy
          severity: error
          businessImpact: operational
          customProperties:
            - property: min
              value: 0
            - property: max
              value: 500
```

Engaging service levels

Service levels usually apply to the entire data product. You will not have a table with a retention of six months and another with a retention of 3 years.

Describing service levels in the data contract needs to be flexible, as more service-level indicators can be created over time. Here is what the definition of the service levels would look like, assuming the end of support is January 1st, 2030, the retention period is 100 years, the dataset was released on October 23rd 2014, and the latency is clearly undefined.

```
slaDefaultColumn: StartDate
slaProperties:
- property: endOfSupport
  value: 2030-01-01T00:00:00-04:00
- property: retention
  value: 100
  unit: y
- property: generalAvailability
  value: 2014-10-23T00:00:00-04:00
- property: latency
  value: -1
  unit: As needed
```

As you can see, service-level indicators are properties, leaving room for extensibility.

You can see many contracts at https://github.com/bitol-io/open-data-contract-standard/tree/main/examples, including excerpts and full ones.

Summary

In this chapter, you learned that trust is really fundamental when it comes to data. It is achieved by following three qualities: having a positive relationship, showing expertise, and being consistent.

Data contracts are key to enabling this trust and in building reliable data products. They can also be used outside of the scope of data products as well, like, for example, documenting data pipelines.

Data Contracts should follow a standard such as ODCS (Open Data Contract Standard) of the Bitol project, hosted by the Linux Foundation AI & Data.

Compared to a non-Data Mesh approach, only creating and maintaining the data contracts are additional work, but they

simplify a lot of the other data engineering team's burden, like documentation and implementation of data quality. Data contracts can help with documentation and they complement tribal knowledge.

Data QoS combines the seven data quality dimensions, as recognized by the EDM Council, with an extensive list of service levels. They can be grouped and organized through a timeline, like a periodic table. Data QoS is an extensible framework that defines the values you can use when implementing data contracts.

Chapter 6. Building your first data quantum

Chapter 7. Aligning with the experience planes

Chapter 8. Meshing your data quanta

Chapter 9. Data Mesh and Generative AI

A NOTE FOR EARLY RELEASE READERS

With Early Release ebooks, you get books in their earliest form—the author's raw and unedited content as they write—so you can take advantage of these technologies long before the official release of these titles.

This will be the 9th chapter of the final book. Please note that the GitHub repo will be made active later on.

If you have comments about how we might improve the content and/or examples in this book, or if you notice missing material within this chapter, please reach out to the editor at sevans@gmail.com.

Generative AI (GenAI) is emerging as a new foundational and generalized technology that is sending shockwaves throughout business and IT. With its unparalleled ability to generate content, insights, and even complete data products, GenAI is poised to revolutionize the very foundations upon which organizations operate.

GenAI represents a seismic shift in the way we perceive and interact with data. Using cutting-edge machine learning techniques, this emerging technology enables systems to create original content, mimic human-like behavior, and generate synthetic data that rivals the real thing.

With GenAI at their disposal, organizations can explore uncharted territories, drive unprecedented creativity, and unlock untapped value hidden within their vast troves of data. This foundational technology is redefining the boundaries of what is possible, shaking the very foundation of business and IT, and propelling organizations into a future brimming with limitless potential.

While still in its early stages, it is probably fair to say this capability is nothing short of a game-changer. So, what is GenAI? Broadly speaking I include several components in GenAI, including Large Language Models, Embeddings (floating point representations of text, or other types of content) that are stored in Vector Databases, and related technology and libraries. So, let's establish a few definitions…

Large Language Models (LLMs)

A large language model is a type of artificial intelligence (AI) system designed to process and interpret written language. What makes it “large” is the immense amount of data it’s trained on during its development. Today’s most powerful LLMs are trained, literally, on virtually all data in the public internet. We’re talking about billions of sentences or text snippets that the model uses to learn and become proficient in understanding how humans communicate through words.

Training an LLM is a complex, time consuming, and costly exercise. During training, an LLM examines massive amounts of text, learning the intricate patterns and structures of language. It becomes an expert at analyzing grammar, context, and the meanings behind words and sentences. This extensive training enables the model to mimic human-like language abilities, letting it communicate in quite a realistic conversational manner.

LLMs have numerous applications, making them incredibly useful and relevant in today’s world. For instance, they can be used in natural language understanding tasks, where they can process and extract information from text, allowing them to perform sentiment analysis, recognize important entities like names or locations, and even classify text into different categories.

LLMs also excel in natural language generation tasks. They produce coherent and contextually relevant text, which is incredibly valuable in tasks like story generation, summarization, or even code writing. Simply put, LLMs are revolutionizing the way we interact with technology, making it more natural and intuitive.

Embeddings

In the context of GenAI, an embedding is a compact and continuous representation of data used to capture the essence or latent features of the input data. The goal of embeddings in generative AI is to transform high-dimensional data into a lower-dimensional space, where each point in the embedding space corresponds to a specific data sample or object.

Embeddings in GenAI are essential because they enable the models to learn meaningful representations of the data, capturing important patterns and relationships. By learning a compact representation, generative models can generate new data that closely resembles the training data, allowing them to create realistic and diverse samples in various domains, such as images, music, text, and more.

Vector Databases

A vector database refers to a collection of vector representations, or embeddings, that encode various attributes, features, or characteristics of data. These vectors represent embeddings that encode various features or attributes of the data.

The super power of a vector database is to efficiently perform “nearest neighbor search.” Imagine you have a massive dataset with thousands or even millions of data points, and you want to find the most similar or closest (“nearest neighbors”) data points to a specific query point. The vector database can quickly identify the data points that are most similar to the query point based on their vector representations.

Beyond nearest neighbor search, a vector database also offers the benefit of efficient retrieval and manipulation of data. By storing data in vector representations, we can work with lower-dimensional and continuous representations of the original data. This can speed up computation times and allow us to perform tasks like dimensionality reduction, data clustering, and classification more effectively.

Moreover, vector databases facilitate diversity and sampling in generative AI tasks. They enable the generation of novel and unique data samples by sampling vectors from the database

and using these samples as “context” to improve our interactions with LLMs.

Challenges

Nevertheless, despite GenAI’s seemingly great power, there are still gaps and limitations. First, there are clear data gaps where the LLM is unable to answer queries in a realistic manner. This is actually a very practical consideration: simply put, the cost and time involved in training LLMs is extensive, measured in the tens and hundreds of millions of dollars and many months of time. This means that only the largest companies—today, only the large internet giants—are able to make the investment necessary to create a competitive LLM.

So, practically speaking, there is a cut-off time when a LLM’s training is completed and any information, data, or events after that cut-off period is omitted which means the data they learn from might be outdated by the time the model is released. And language is dynamic and current events constantly reshape the world we live in, so LLMs struggle to keep up with current trends, slang, or emerging concepts, leading to inaccuracies or outdated responses in real-world scenarios.

Another limitation is the potential bias in the training data. Since LLMs learn from publicly available text data, they might unintentionally absorb biases present in the data sources. For instance, if a specific website or forum has a biased view on certain topics, the model could reflect that bias in its outputs. For example, gender bias where models were trained on gender stereotypes like “traditional jobs" which may emphasize particular gender. As a result, LLMs might not always provide fully objective or balanced answers, which could be problematic, especially in sensitive or controversial topics.

Additionally, LLMs have been criticized for their inability to fully understand context or maintain coherent conversations over extended interactions. While they can generate text that seems impressive at first glance, they may struggle to maintain context and relevance in more extended conversations, leading to responses that are nonsensical or irrelevant. This limitation could be attributed to the models’ lack of true understanding and long-term memory.

Next, virtually all enterprise data—information about customers, products, employees, etc—is considered private, sensitive, or confidential, meaning that LLMs today are not trained on enterprise data. So, just like data, information, or events after the training cut-off period, LLMs are blind to

enterprise data. Consequently, LLMs don't have access to this proprietary information, which might limit their ability to provide precise answers or insights in certain enterprise-specific domains.

Last, the lack of access to private data can impact the performance of LLMs in specialized domains. For example, a company working on cutting-edge research or proprietary technology might require AI models that have an in-depth understanding of their domain-specific jargon and knowledge. Since LLMs lack exposure to such private data, they might not be as effective in these specialized contexts.

But most of the headlines today related to GenAI's capability using data that originates beyond the four walls of the enterprise. However, almost all large language models were trained on the data riches of the whole internet, but, literally, know nothing about the data that exists within an enterprise.

Now, the fusion of Data Mesh and GenAI give organizations a tantalizing new opportunity: Use Data Mesh as the foundational data platform that makes it easy for Generative AI to find, consume, share, and trust enterprise data. And together, Generative AI and Data Mesh make it easy to create insights. Simply put, Data Mesh supercharges Generative AI.

So let’s dive a bit deeper.

Data Mesh and Generative AI

The integration of GenAI with Data Mesh architecture is a thrilling advancement that promises to reshape the landscape of data management. By leveraging the unparalleled capabilities of GenAI within the decentralized framework of Data Mesh, organizations are poised to unleash a torrent of transformative insights and revolutionize their data-driven decision-making processes!

Where GenAI empowers enterprises to autonomously generate content, insights, and even complete data products, Data Mesh makes the data required by GenAI easy to find. Consume, share, and trust.

So, imagine the ability to deploy GenAI models across various domains within an organization, enabling stakeholders to effortlessly generate synthetic data, simulate complex scenarios, and extract profound patterns that fuel strategic initiatives with an unprecedented fervor.

The synergy between GenAI and Data Mesh holds immense potential to reshape the enterprise data landscape. With the

incredible power of generative AI to consume enterprise data, privacy concerns are alleviated propelling innovation and experimentation to new heights and directions.

Furthermore, Data Mesh's decentralized nature perfectly complements the autonomous capabilities of generative AI systems, empowering individual data product teams working with GenAI enabled data scientists to wield LLMs customized to enterprise-specific data that align precisely with their unique requirements and objectives.

An Architecture for Generative-AI

Usually there is plentiful data documentation, or content, with an enterprise – data definitions, schemas, and other metadata, as well as developer documentation. There are two problems with this data: first the content is available in numerous different structures (CSV, database schema), formats (PDF, text files, documents), and media (files, URLs) which makes it quite difficult to consume and understand. Second, GenAI has no understanding of enterprise content because it was not trained on it.

So, how can these challenges be addressed? How can the various different forms of content be "normalized" into a

usable form? And, ideally, how can the semantic richness of the content be maintained as it is normalized? Moreover, how can we make this internal enterprise content known to GenAI so that it can provide meaningful responses to user requests and queries?

Our GenAI enabled Data Mesh architecture, illustrated in Figure 9-1, addresses both of these challenges. Our generalized architecture uses GenAI to, first, normalize content while maintaining its semantic richness, while, second, also allowing users to make requests and queries upon internal enterprise data.

Let's explore our generalized architecture for GenAI.

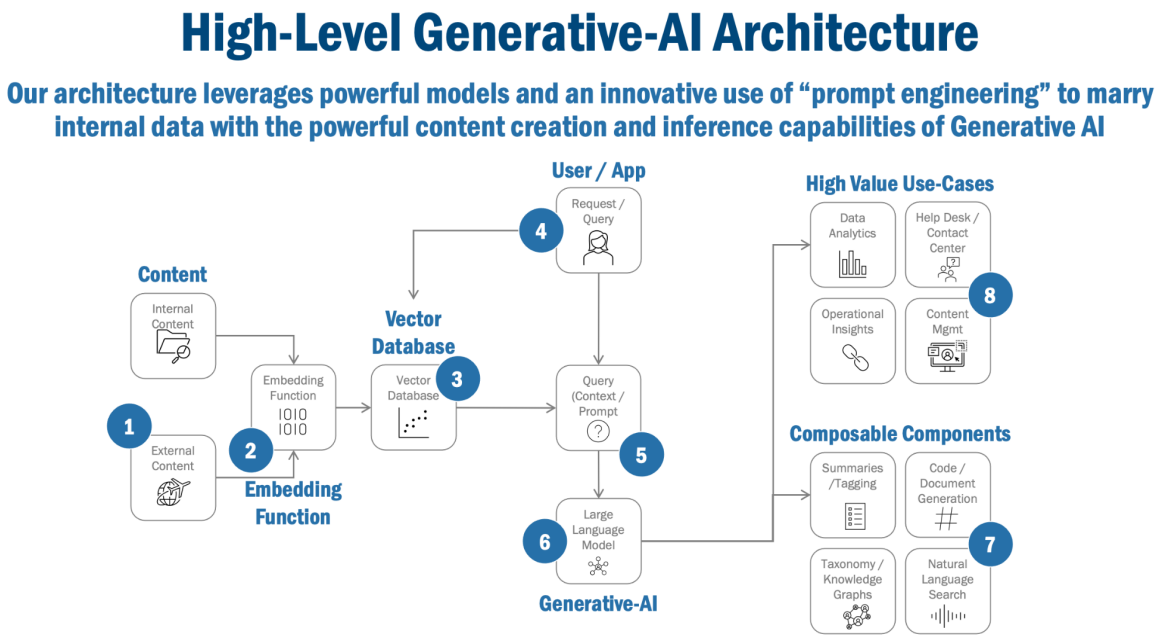


Figure 9-1. , High-Level Generative-AI Architecture

Our architecture combines many GenAI components—large language models, embeddings, and vector databases—as well as sophisticated prompt engineering to generate text, content, and insights. We will use this architecture to "GenAI enable" our Data Products and Data Mesh. But more on that later.

Rather, first let's define the components in our architecture. First, *Content* (#1 in diagram) that is consumed by and used by our architecture as the basis for addressing requests and answering queries. This content may come from internal or external sources and documents such as data descriptions, developer documentation, tutorials, and guides that are related to the queries and requests we have. Today, tools are available

to process just about any data types including text, CSV (comma separated values), Microsoft Office documents (practically, any textual data type) and most access methods (raw files, storage buckets, URLs).

Next is the *Embeddings Function* (#2), which converts textual content into vector representations, typically called embeddings. The most interesting and useful thing about embeddings is that modern techniques allow these embeddings to have strong and rich relationships to the actual semantics of original content.

Embeddings are stored in a *Vector Database* (#3) while providing sophisticated search capabilities above and beyond traditional databases. Of particular interest is that vector databases offer “nearest neighbor” search which finds vectors that are most similar to an input vector—now, recognizing that the embeddings are accurate representation of the semantics of the original content, this allows very capable searches based upon the semantics of the input query. The power of this is immense—consider an arbitrary search for, say, the word “dog”—older techniques would find exact matches (all instances of “dog”), or perhaps fuzzy matches (words that have overlapping text such as “dock”), but using embeddings and vector database similarity searches, we find that “dog” would

match semantically similar words such as, for example, “cat” (it is an animal) or “german shepherd” (a type of dog).

Once embeddings are stored in the vector database, a *User Request/Query* (#4) can be issued (in our case, perhaps, data scientists or data analysts) that request data or ask questions about our Data Products.

Next, *Prompts (#5)* are a combination of the user query and a “context”. The context is particularly important as it provides guidance and, in some cases constraints, on how GenAI responds to user requests / queries. But where does the context come from? Well, the context is the output from a search from our vector database. But, again, what is the actual search request presented to the vector database? Well, the search is the user query. But why the original query? Well, the original query, when converted to an embedding, provides the semantics of the user request—so, when we search the vector database we are actually searching for items that are semantically similar to our query. And with some basic prompt engineering, the original query is combined with the context to guide GenAI as it responds to a user request/query.

Large Language Models (#6) respond to prompts. The so-called super-power of these models—the thing that has only now been

achievable—is that they respond to user requests / queries in a a compelling, human-like, and conversational way to based upon provided data combined with the knowledge gleaned from being trained on the vast riches of the internet. And by using our architecture, we allow the models to also respond to our data, the information in our Data Products.

What do we do with this powerful capability? Well, first, we can create *Composable Components* (#7) that can be used to build applications. While the types of composable components is only limited by our imagination, there are some obvious ones to consider including:

Summaries and Tags

Documentation from Data Products—in any form and by any access mechanism—and ingested in our GenAI architecture to summarize vast sets of documentation into summaries and tags suitable for use in a catalog of Data Products.

Taxonomy and Knowledge Graphs

These artifacts provide insights that aid in understanding how Data Products (and their data) are interconnected and how their data flows.

Code/Document Generation

While strictly speaking this may be initially considered an aid to developers, this capability nevertheless is much more than that. It is quite common and relatively simple to generate prompts which use our GenAI architecture to act upon user, developer, and data documentation, to create OpenAPI specifications, JSON schemas, as well as software fragments used by Data Product developers as well as Data Product consumers. And with some creative prompts we can generate reports, analyze data, identify trends, and make recommendations and predictions based upon data that is unique to the Data Products in our enterprise.

Natural Language/Semantic Search

Queries issued in natural language (in contrast to older SQL style queries, or keyword search) are converted into embeddings which retain the semantics of the query. In our architecture, a vector database of embeddings is a repository of rich semantically searchable content, so any query that uses the vector database “nearest neighbor” search capabilities will return content that is semantically similar to the search query.

Lastly, High Value Use Cases (#8) can be built using raw GenAI capabilities, or better yet, accelerated through the use of our composable components. Again, while limited only by our imagination, several useful solutions can be considered:

Data Analytics and Reporting

GenAI composable components can be integrated into pipelines to create—at scale—enterprise reports and analysis.

Operational Insights

By ingesting operational documentation—support documentation, error logs, or usage statistics—help desk capabilities can be supercharged with Natural Language Search.

Help Desk / Contact Center

In our architecture, content—documentation, operations run books, training material, for example—is converted to embeddings, while retaining their semantic richness of the original content—can be searched using our Natural Language Search components making it easy for Help Desk and Contact Center to find information required by users. In addition, with our document generation

composable components, responses to queries can easily be generated and customized for specific user circumstances.

Content Management

Our GenAI architecture has composable components to create knowledge graphs, taxonomies, and tags for content (web sites, catalogs, internal document stores) which is useful for content categorization, management and governance.

Enterprise Data Mesh and Generative AI

Clearly GenAI offers powerful new capabilities for an enterprise. But how does it power an enterprise's Data Mesh? Several opportunities come to mind.

First, Data Product On-boarding—making it known and discoverable in the Data Mesh—is a challenging activity for several reasons. First, information such as descriptions, metadata, taggings, and potentially knowledge graphs and taxonomies, is hard to find. Even when this information is found, there is little incentive to get those knowledgeable about the data to provide the necessary information. And, even with

an incentivized and motivated group, it takes significant effort and time to create the necessary content in the required format.

So, how can GenAI composable components address these challenges? Simply put, our GenAI architecture makes it easy to gather the necessary information, format it in a usable and searchable form. And all of this is done with minimal effort from busy experts.

But, how does this work? First, since our GenAI architecture can ingest any form of data using just about any access mechanism, we can easily acquire vast amounts of content about the data for our Data Product. Next, once acquired, our Summarization and Tagging composable component can capture the semantic richness of the content and generate consistent sets of content summaries.

Next, with on-boarded data products, Data Products now can be easily made discoverable. To do this, we can load our summarizations and tags into the Data Product Registry (see Chapter 3 for more information) which acts as the catalog for an enterprise Data Mesh. Now, any user can easily find the data products based upon semantically rich summaries and tags.

So, now that we have rich content in our Data Product Registry, we make it searchable using our Natural Language Search composable components. Now, all of the semantically rich information about Data Products is available using natural language and semantically rich search. No more SQL. No more guessing key words. Just simple, intuitive questions.

Applying Generative-AI to Climate Data Inc

So, how do these components interact in our Climate Data Inc scenario?

GenAI can be immensely beneficial for managing vast amounts of climate data in Climate Data Inc. Let's explore take a look at a few use cases:

Climate Data Search

GenAI's natural language / semantic search capabilities offer a significant advantage in finding climate data efficiently. With the ability to understand the intent behind user queries and interpret complex climate-related terms, GenAI enables more accurate and relevant search results. Users can pose questions in plain language, and the AI model can process these queries, analyze their meaning, and retrieve climate data that matches the context of the question. And by integrating with knowledge

graphs and tagged data, generative AI provides comprehensive search results, including related datasets, geographical locations, time periods, and specific climate variables. This simplified search process enhances data accessibility, fosters a deeper understanding of climate patterns, and facilitates informed decision-making and advancements in climate science and policy.

Climate Data Summarization

Climate data often consists of large and complex datasets, including weather observations, satellite imagery, and climate models. GenAI can be employed to summarize this data, extracting key insights, patterns, and trends from the voluminous information. By using both extractive and abstractive summarization techniques, GenAI can generate concise summaries of weather events, climate change trends, and extreme weather occurrences. These summaries aid decision-makers, scientists, and policymakers in quickly understanding the critical aspects of the climate data without having to delve into the minutiae.

Climate Data Tagging

GenAI can assist in tagging climate data, making it easier to categorize and organize the information. By identifying key variables, regions, and temporal aspects within the data, the LLMs can create relevant tags or keywords. For instance, the model can tag weather data with attributes like temperature, precipitation, humidity, and location details. These tags help in efficient data retrieval and analysis, making it simpler to compare and correlate different climate variables and their impacts over time.

Climate Data Knowledge Graphs

GenAI can be utilized to create knowledge graphs that represent the relationships between various climate-related entities and concepts. For instance, a knowledge graph can connect climate data, geographical locations, climate models, and research papers, showcasing how different factors interact with each other. This interconnected representation allows scientists and researchers to navigate the complex web of climate data and discover potential causal relationships and patterns.

Code Generation for Climate Data Consumers

With the vast amounts of climate data being collected from diverse sources, integrating this data into applications and analytical tools can be time-consuming and challenging. GenAI can automate code generation for data integration, creating code snippets that fetch, process, and preprocess climate data from various repositories. This automation not only saves time but also ensures consistency and accuracy in the integration process.

Summary

By leveraging GenAI's capabilities in summarization, tagging, knowledge graph creation, and code generation, managing vast amounts of climate data in our Data Mesh becomes much more efficient and insightful. It facilitates faster decision-making, scientific research, and policy analysis, contributing to a better understanding of climate patterns, impacts, and potential mitigation strategies. And, as climate data continues to evolve, GenAI can adapt and update its models, ensuring that the climate data management process remains relevant and effective over time.

Part III. Teams, operating models, and roadmaps for Data Mesh

With a basic Data Mesh in place, how do you successfully set up the teams, operating model, and roadmap required to establish, nurture, and grow your Data Mesh? You will also discover a maturity model that will enable you to measure your progression.

Chapter 10. Running and operating your Data Mesh

Chapter 11. Implementing a Data Mesh Marketplace

Chapter 12. Implementing Data Mesh Governance

Chapter 13. Running your Data Mesh Factory

Chapter 14. Defining and Establishing the Data Mesh Team

A NOTE FOR EARLY RELEASE READERS

With Early Release ebooks, you get books in their earliest form—the author's raw and unedited content as they write—so you can take advantage of these technologies long before the official release of these titles.

This will be the 14th chapter of the final book. Please note that the GitHub repo will be made active later on.

If you have comments about how we might improve the content and/or examples in this book, or if you notice missing material within this chapter, please reach out to the editor at sevans@gmail.com.

Data Mesh can be metaphorically described as an ecosystem of data. Just like a natural ecosystem, Data Mesh consists of interconnected components that work together to create a thriving environment for data-driven value creation. But as importantly, Data Mesh is an ecosystem of interacting teams - teams that consume data platforms, teams that publish data

platforms, teams manage data platforms, and teams that govern data platforms.

So, while the technical aspects of data mesh receive a lot of attention, it's clearly important to acknowledge the significant organizational considerations involved. In this chapter, I delve into the challenges of implementing a data mesh and highlight the crucial role of organizational changes in this journey.

Team Topologies in Data Mesh

First, let's start with the core responsibilities within the Data Mesh ecosystem. Clearly, the primary responsibility of the various Data Mesh teams is to safely, efficiently, and effectively manage data. Of course, there is more:

Data Governance

> Implementing a robust data governance framework is crucial for Data Mesh. This framework should define data ownership, data quality standards, data privacy policies, and data access controls. It establishes guidelines for data product teams to ensure consistency, compliance, and accountability across the organization.

Decentralized Data Ownership

Transitioning from a centralized data ownership model to a decentralized one is a fundamental aspect of Data Mesh. This involves redefining ownership and accountability for data products, empowering domain experts, and enabling cross-functional teams to manage their own data products.

Data Platform as a Service

Establishing a data platform as a service is essential to support data product teams. This platform provides the necessary infrastructure, tools, and services to facilitate data ingestion, storage, processing, discovery, and sharing. It should enable self-service capabilities for data product teams and promote interoperability and scalability.

Standards and Best Practices

Implementing governance mechanisms within Data Mesh is crucial to ensure alignment and collaboration across teams. This includes defining standards, protocols, and best practices for data product development, sharing knowledge and learnings through communities of practice, and fostering collaboration between data product teams and other stakeholders.

Data Mesh Architecture and Infrastructure

Organizations need to design and implement the technical architecture and infrastructure required to support the data mesh. This may involve leveraging cloud technologies, distributed data storage systems, data pipelines, and scalable analytics platforms. The architecture should enable data product teams to work independently while ensuring data consistency and integration.

Data Democratization and Data Literacy

Promoting data democratization and fostering data literacy across the organization are critical aspects of Data Mesh. This involves providing training and education programs to empower employees to leverage data effectively, promoting a data-driven culture, and ensuring that data is accessible to those who need it.

Metrics and Monitoring

Defining relevant metrics and establishing monitoring mechanisms are essential to track the performance, usage, and value of data products within the data mesh. This enables continuous improvement, identifies areas for optimization, and helps measure the impact of data products on business outcomes.

To establish effective data product teams within Data Mesh, I leverage the concept of team topologies adapted from Matthew Skelton and Manuel Pais in their Team Topologies book. As shown in Figure 9.1, the core team in our discussion is the Data Product Teams (Pais/Skelton's Stream-aligned teams) that own, manage, and govern data products, however, we will also touch briefly upon the Data Platform and Data Enabling teams that support Data Product teams.

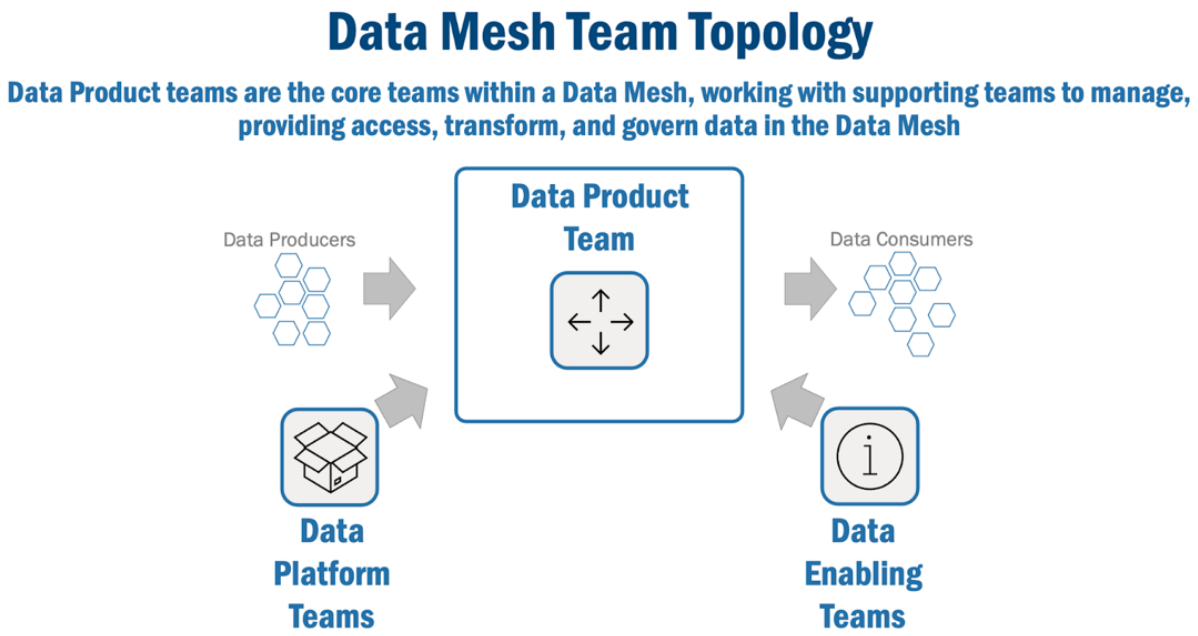


Figure 14-1. Data Mesh Team Topology

In figure 9-1, we show three types of teams: the "Data Product" team, the "Data Platform" team, and the "Data Enabling" team. So, what do each of them do?

Data Product teams are the core of the Data Mesh team ecosystem. They are the experts in consuming data from data providers, transforming data to deliver business value, and making it available to data consumers. In a Data Mesh ecosystem there are many data product teams, each taking on the responsibility of end-to-end delivery of services related to a specific data product. A data product team interacts with groups that create, manage, and consume as well as platform and enabling teams that provide technical and support capabilities.

In our “ecosystem” metaphor, data products are analogous to different species within the ecosystem. Each Data Product team acts as a unique species, specializing in a specific domain or data-related service. Just as diverse species contribute to the overall biodiversity and functionality of an ecosystem, data products contribute their distinct data capabilities and expertise to the data mesh ecosystem.

There are two key teams that support the Data Product teams: Data Platform teams and Data Enabling teams.

Let’s touch upon Data Platform teams first. Where Data Product teams are experts in consuming data technology, the Data Platform team are experts in managing data technology. The

Data Platform teams play a crucial role in supporting and enabling the work of other Data Product teams within the Data Mesh ecosystem. These teams focus on providing the necessary tools, utilities, and technical services that make it easier for data product teams to perform their tasks efficiently and effectively. Platform teams act as a central resource, offering shared services and capabilities that can be utilized by multiple teams across the organization.

The primary goal of Data Platform teams is to remove any friction or barriers that other teams may encounter during their development and delivery processes. They build and maintain the underlying infrastructure, frameworks, and services that streamline the development, deployment, and operation of software products or services. By providing these “X-as-a-Service” capabilities, platform teams allow other teams to focus on their specific areas of expertise without having to reinvent the wheel.

The scope of Data Platform teams can vary depending on the organization’s needs and priorities. They may include teams specializing in areas such as cloud infrastructure, APIs, security, networking, or any other technical domain that is critical for supporting the organization’s software development efforts. Data Platform teams collaborate closely with other teams,

understanding their requirements and continuously improving the services they offer to ensure smooth and efficient operations.

Data Platform teams offer “X-as-a-Service” capabilities, but, in our case, services related to data including data storage, processing, security, and integration, to make it easier for data product teams to develop and operate their data products effectively.

Continuing with our ecosystem metaphor, the Data Platform teams can be seen as the infrastructure that supports and nourishes the ecosystem. They provide the necessary tools, services, and technical capabilities akin to the fertile soil, clean water, and favorable climate that enable the growth and sustainability of the ecosystem. These platform teams ensure that the data products have the necessary resources and support to operate effectively and deliver value.

Last, but definitely not least, are Data Enabling teams. They play a vital role by providing consultative support and expertise to Data Product teams. They help address obstacles, offer guidance, and promote best practices in data management and governance.

Data Enabling teams are specialized groups within an organization that provide support and expertise to other teams in order to overcome obstacles and address specific needs. These teams act as consultants or advisors, offering guidance, resources, and solutions to help teams navigate challenges and achieve their goals.

The role of Data Enabling teams is to identify and understand the unique requirements and constraints faced by other teams. They collaborate closely with these teams, working in short bursts or on a project basis to provide targeted assistance. Enabling teams bring their expertise and knowledge to bear on specific problems or areas where additional support is needed.

Data Enabling teams can take different forms depending on the organization and its specific needs. They may include steering groups, enterprise governance and architecture teams, training groups, or any other specialized teams that can offer insights and assistance in specific domains. These teams typically have deep knowledge and experience in their respective areas and can provide valuable guidance, best practices, and resources to help teams succeed.

By leveraging the expertise of Data Enabling teams, Data Product teams can benefit from specialized support and

knowledge without having to build the same capabilities within every individual team. Data Enabling teams help foster collaboration, knowledge sharing, and innovation across the organization by providing targeted support to teams facing challenges or pursuing opportunities.

Data Enabling teams, in our metaphor, can be likened to symbiotic relationships within an ecosystem. They act as facilitators and collaborators, providing guidance, expertise, and support to the data product teams. Similar to how certain species in an ecosystem assist and rely on each other for survival, enabling teams help data product teams overcome challenges, foster best practices, and promote the overall health and success of the data mesh ecosystem.

Still, the Data Product team is the core team with a Data Mesh, so let's dig a bit deeper here. What does a Data Product team do? What are the key roles on the team? And what benefits and challenges do they experience?

The Data Product Team

Data Product teams are self-contained, autonomous units within an organization responsible for the end-to-end delivery of data products and services in the Data Mesh. These teams

operate according to Data Mesh principles that emphasizes decentralized data ownership, distributed architecture, and a federated governance model. Data product teams are crucial for effectively managing and delivering data-driven value at scale.

A Data Product team has a clear scope and boundaries, typically centered around a specific database, set of tables, or files. They are accountable for all aspects of the Data Product lifecycle, including data ingestion, consumption, discovery, observability, and ensuring its overall success in delivering value to the organization.

But most importantly, each Data Product team works independently and has the autonomy to make decisions regarding their Data Products. And it is this autonomy and independence that allows for faster decision-making and shorter feedback loops which are essential for delivering data-driven solutions efficiently.

Moreover, Data Product teams are typically organized around specific business domains or product areas, which enhances their understanding of the domain and enables them to make informed decisions. Similarly, a Data Product team, with its dedicated focus on a particular Data Product or service, can

develop deep expertise and a comprehensive understanding of the data and its implications within the specific domain.

Data Product teams interact with various other teams within the data mesh. Producer teams, which manage the source of data, collaborate with data product teams to ensure smooth data ingestion. Consumer teams, on the other hand, access and utilize the data offered by the data product team for various analytical purposes. Platform teams provide essential “X-as-a-Service” capabilities to support the data product team’s data ingestion, consumption, and sharing processes. Enabling teams assist the data product team in overcoming short-term obstacles or addressing specific needs.

Key Roles and Responsibilities

The structure of a Data Product team, shown in figure 9-2, can vary depending on the specific requirements of the Data Product they are working on. The team is led by a Data Product owner who holds accountability for the success of the data product. Other roles within the team may include metadata management, data management and security, consumption services, ingestion services, and release management. Each role contributes to different aspects of the data product’s lifecycle

and ensures its smooth operation and delivery of value to the organization:

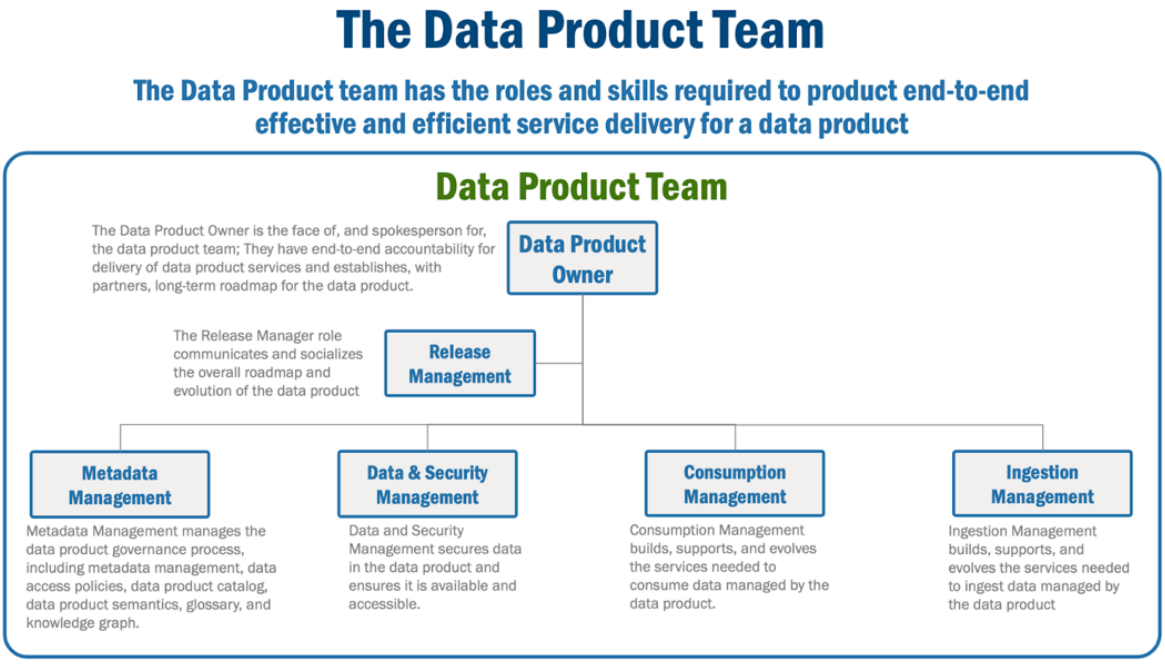


Figure 14-2. The Data Product Team

It's important to note that while these roles are commonly found within data product teams, the specific structure and composition can vary depending on the organization and the nature of the data product being developed. The size of a data product team varies depending on its particular purpose and objective. In some cases, Data Product teams may be quite small perhaps where scope is limited, but in other cases may be somewhat larger. However, in most cases, the old AWS maxim of a "two pizza team" (about 10-12 people) is probably a practical maximum.

A Data Product team can be metaphorically described as an orchestra, where each member plays a unique instrument and contributes to the collective performance. Just like in an orchestra, a data product team consists of individuals with specialized skills and roles, working together harmoniously to deliver valuable data-driven services.

In this metaphor, the Data Product team leader can be seen as the conductor of the orchestra. They set the vision, provide guidance, and ensure that all team members are aligned and working towards a common goal. Like a conductor, they coordinate and bring together the diverse talents and expertise of the team members to create a cohesive and impactful performance.

The different roles within the Data Product team can be compared to various musical instruments. Each role has its own specific responsibilities and skills, similar to how each instrument has a distinct sound and purpose in an orchestra. Whether it's the Data Product owner, metadata manager, consumption services manager, or ingestion services manager, each member contributes their expertise and skills to the overall symphony of Data Product delivery.

Just as musicians in an orchestra rehearse and refine their performance, a Data Product team collaborates, iterates, and continuously improves their work. They practice together, refine their processes, and strive for excellence in delivering high-quality data services to their stakeholders, just like musicians aim for a flawless and captivating performance.

The orchestration metaphor highlights the importance of coordination, collaboration, and synchronized efforts within a Data Product team. Each member has a crucial part to play, and it is through their collective expertise and collaborative spirit that they create a harmonious Data Product that delivers value to the organization.

Now, let's take a look at how an individual Data product team is organized and explore key team roles and responsibilities.

Data Product Owner

The Data Product team is led by a "Data Product Owner" who has overall accountability for the success of the data product. This role is responsible for setting the direction and roadmap of the data product, acquiring funding, and liaising with stakeholders and other teams involved.

Data Product Owner's responsibilities encompass various aspects of strategic planning, stakeholder management, and product delivery. Here are some of the most important responsibilities and skills of a Data Product Owner:

Strategic Direction

The Data Product Owner is accountable for setting the strategic direction of the data product. They need to understand the business objectives, customer needs, and market trends to define the vision, goals, and roadmap for the data product. They should align the data product's objectives with the overall organizational strategy and ensure its continued relevance and value.

Stakeholder Management

The Data Product Owner is responsible for identifying and engaging with stakeholders who interact with the data product. They need to establish effective communication channels, build relationships, and gather feedback to understand the needs and expectations of stakeholders. They should act as the main point of contact for stakeholders, addressing their concerns, and ensuring that the data product meets their requirements.

Cross-Functional Collaboration

The Data Product Owner needs to collaborate with various teams and individuals across the organization. They work closely with data engineers, data analysts, domain experts, and other stakeholders to define and deliver the data product. Collaboration skills are essential for fostering a culture of teamwork, encouraging knowledge sharing, and ensuring alignment between different teams and functions.

Product Ownership

The Data Product Owner is responsible for its overall success. This includes defining the product strategy, prioritizing features and enhancements, and managing the product backlog. They should continuously monitor and evaluate the performance of the data product, making data-driven decisions to optimize its value, usability, and impact.

Domain Knowledge

Data Product Owners should have a deep understanding of the domain in which the data product operates. This includes knowledge of relevant industry trends, business processes, and data-related challenges within the domain. They need to be well-versed in the domain-specific

terminology, data requirements, and analytics needs to effectively guide the development and delivery of the data product.

Leadership and Management

Data Product Owners need strong leadership and management skills to drive the success of the team and the data product. They should inspire and motivate team members, provide guidance and support, and foster a collaborative and innovative work environment. Effective management skills, including resource allocation, prioritization, and conflict resolution, are essential for ensuring the efficient delivery of the data product.

Continuous Learning and Adaptability

The data landscape is constantly evolving, and Data Product Owners need to stay updated with the latest industry trends, technologies, and best practices. They should be open to learning, adapt to changing circumstances, and embrace new approaches to data product development and delivery. This includes being proactive in exploring emerging technologies, attending industry conferences, and participating in relevant training and development opportunities.

By fulfilling these responsibilities and possessing these skills, a data product team owner can effectively guide the development and delivery of data products within a data mesh, ensuring their alignment with business objectives and delivering value to stakeholders.

Release Manager

The Release Manager manages, communicates, and socializes the overall roadmap and evolution of the data product. They work closely with all members of the team to integrate changes into the release process and effectively communicate updates and changes to stakeholders and users of the data product.

The Release Manager plays a crucial role in coordinating and managing the release process of data products within a data mesh. Their responsibilities revolve around ensuring the successful deployment, communication, and adoption of data product releases. Here are some of the most important responsibilities and skills of a data product release manager:

Release Planning and Coordination

The Release Manager is responsible for planning and coordinating the release of data products. They work closely with the data product team owner, development

teams, and other stakeholders to define release schedules, establish release criteria, and coordinate the release process. They ensure that releases are well-planned, aligned with business objectives, and meet quality standards.

Release Management Process

The Release Manager develops and implements an effective release management process. This includes defining release workflows, versioning strategies, and release documentation. They establish and enforce release management best practices, ensuring that proper testing, validation, and approval processes are in place before releasing data products to production environments.

Communication and Stakeholder Management

Effective communication is key to successful release management. The Release Manager is responsible for communicating release plans, progress, and impacts to stakeholders, including data product team members, business users, and technical teams. They should have strong interpersonal and communication skills to facilitate clear and timely communication, manage

expectations, and address any concerns or issues related to the release.

Change Management and Risk Mitigation

The Release Manager identifies potential risks and impacts associated with data product releases and develops strategies to mitigate them. They work closely with the data product team, testing teams, and stakeholders to assess the impact of changes, manage dependencies, and ensure smooth transitions during release deployments. They should have a proactive approach to change management and be able to adapt to unexpected situations or challenges that may arise during the release process.

Continuous Improvement and DevOps Practices

The Release Manager plays a role in driving continuous improvement and implementing DevOps practices within the data product release process. They collaborate with development and operations teams to automate release processes, improve efficiency, and enhance overall release quality. They should have a strong understanding of DevOps principles, tools, and methodologies to optimize the release management lifecycle.

Technical Knowledge and Troubleshooting

The Release Manager should have a solid understanding of the technical aspects of data product releases. They should be familiar with deployment strategies, infrastructure requirements, and configuration management. This knowledge allows them to effectively troubleshoot issues, coordinate with technical teams, and ensure successful deployments of Data Products.

Documentation and Reporting

The Release Manager maintains proper documentation of release processes, version control, and release notes. They ensure that accurate and up-to-date information is available to stakeholders regarding the content and impact of data product releases. They also generate release reports, metrics, and key performance indicators (KPIs) to track release performance and identify areas for improvement.

By fulfilling these responsibilities and possessing these skills, aRelease Manager can ensure smooth and successful releases of data products within a data mesh. Their coordination, communication, and technical expertise contribute to the efficient delivery and adoption of data product releases,

ultimately driving value for the organization and its stakeholders.

Metadata and Governance Manager

The Metadata and Governance Manager is responsible for defining, governing, and managing metadata for the data product. They ensure the integrity of data product semantics, maintain the glossary, and manage the knowledge graph related to the data product.

The Metadata and Governance Manager plays a crucial role in managing and governing the metadata associated with data products within a data mesh. Their responsibilities revolve around defining, organizing, and maintaining metadata to ensure its accuracy, consistency, and usability. Here are some of the most important responsibilities and skills of a Metadata and Governance Manager:

Data Governance

The Metadata and Governance Manager is responsible for establishing and enforcing metadata governance policies and processes. They define standards, guidelines, and best practices for metadata management, ensuring that metadata is captured, documented, and maintained

consistently across data products. They play a key role in ensuring data quality and compliance with regulatory requirements.

Data Contracts

The Metadata and Governance Manager defines the data contracts that provide the necessary information to support data access, transformation, and data quality management.

Metadata Design and Architecture

The Metadata and Governance Manager designs and implements metadata models and structures that capture the relevant information about data products. They define metadata attributes, relationships, and classifications that enable effective data discovery, understanding, and usage. They should have a strong understanding of data modeling techniques and metadata standards (such as Dublin Core, W3C PROV, etc.).

Metadata Documentation and Cataloging

The Metadata and Governance Manager is responsible for documenting and cataloging metadata in a centralized metadata repository or catalog. They ensure that

metadata is properly described, tagged, and indexed to facilitate easy search, retrieval, and understanding of data products. They may use metadata management tools or data cataloging platforms to support these activities.

Data Lineage and Impact Analysis

The Metadata and Governance Manager establishes processes and tools to capture and maintain data lineage information. They track the origins, transformations, and dependencies of data products to enable impact analysis and traceability. They should have a strong understanding of data integration and transformation processes to accurately capture and represent data lineage.

Data Quality and Integrity

The Metadata and Governance Manager ensures the quality and integrity of data in the Data Product by performing data quality assessments, implementing data validation rules, and resolving any issues or inconsistencies in metadata. They collaborate with data owners, data stewards, and other stakeholders to improve the quality and completeness of metadata, ensuring its reliability for data discovery and decision-making.

Collaboration and Stakeholder Management

The Metadata and Governance Manager collaborates with various stakeholders, including data product owners, data engineers, data analysts, and business users. They engage with these stakeholders to understand their metadata requirements, gather feedback, and ensure that metadata supports their data needs effectively. They should have excellent communication and interpersonal skills to foster collaboration and build relationships with stakeholders.

Knowledge of Metadata Standards and Technologies

The Metadata and Governance Manager should have a solid understanding of metadata standards, such as the aforementioned Dublin Core, W3C PROV, or industry-specific metadata frameworks. They should also be familiar with metadata management tools, data cataloging platforms, and other relevant technologies used for metadata management. Staying updated with emerging trends and advancements in metadata management is essential for effectively performing their role.

Continuous Learning and Adaptability

The metadata landscape is constantly evolving, with new data sources, technologies, and metadata management

practices emerging. Like other roles in the Data Product team, the Metadata and Governance Manager should have a passion for continuous learning and adaptability to stay updated with industry trends and incorporate new metadata management techniques into their practice. They should be open to exploring innovative approaches and leveraging new tools and technologies to improve metadata management processes.

The ownership of a data catalog can vary depending on the organizational structure and governance model. Typically, it is owned and managed by the data governance or data management team within the organization. This team is responsible for ensuring the accuracy, consistency, and availability of data catalog information. They collaborate with data producers, data product teams, and other stakeholders to collect and maintain the necessary metadata, data lineage, and documentation in the catalog.

The data catalog owner's responsibilities include overseeing the catalog's maintenance, defining data catalog policies and standards, establishing data governance processes, and ensuring data quality and data security within the catalog. They collaborate with data consumers and data producers to understand their needs and provide them with a user-friendly

interface to search, discover, and access the data assets they require. The data catalog owner also plays a critical role in promoting data catalog adoption, training users on its usage, and continuously improving its functionality based on feedback and evolving business requirements.

By fulfilling these responsibilities and possessing these skills, a data product metadata manager can effectively govern, manage, and utilize metadata within a data mesh. Their expertise in metadata governance, documentation, and collaboration contributes to the overall success and usability of data products, supporting data discovery, understanding, and decision-making across the organization.

Metadata management and governance within a data mesh can be metaphorically compared to a marketplace, where information and knowledge are traded and regulated. In this metaphor, the metadata serves as the valuable goods and services available in the marketplace, and metadata management and governance act as the stewards and regulators of this marketplace.

Imagine the metadata as various products displayed in different stalls within a bustling marketplace. Each stall represents a different aspect of metadata, such as data semantics, glossary,

knowledge graph, and data lineage. Metadata managers oversee these stalls, ensuring that the metadata is accurate, up-to-date, and aligned with the organization's standards and guidelines.

Just as market regulations ensure fair trade practices and protect the interests of buyers and sellers, metadata governance establishes rules, policies, and frameworks to ensure the proper management, usage, and sharing of metadata. Metadata managers act as the regulators, monitoring compliance, resolving disputes, and enforcing standards to maintain the integrity and consistency of the metadata within the data mesh ecosystem.

Data consumers, like customers in a marketplace, rely on the metadata to discover, understand, and effectively utilize the available data products. Metadata management and governance ensure that the metadata is easily accessible, well-organized, and provides accurate and relevant information about the underlying data assets.

Through the marketplace metaphor, we can grasp the vital role of metadata management and governance in establishing a well-functioning data mesh. They facilitate the discovery, accessibility, and trustworthiness of metadata, allowing data

consumers to navigate and leverage the diverse range of data products within the ecosystem effectively.

Data and Security Manager

The Data and Security Manager has responsibility for the overall architecture of the Data Product. The collaborating with other team members to design and maintain a scalable and efficient data architecture that supports the organization's data needs. The Data and Security Manager has skills that include:

Data modeling

The Data and Security Manager can conceptualize complex business requirements and translate them into a logical and structured representation. They analyze business processes, identify relevant entities, attributes, and relationships, and capture them in a conceptual data model and have the ability to simplify real-world complexities into manageable data structures.

Data storage and retrieval mechanisms

Effective data storage and retrieval management is crucial for maintaining data integrity, supporting efficient data analysis, and enabling timely decision-making within an organization. The Data and Security Manager is an expert

in efficiently storing, organizing, and retrieving data within an organization. They ensure that data is stored in a secure and scalable manner, allowing for quick and reliable access when needed. This role also addresses data redundancy, backup and recovery procedures, data compression techniques, and leveraging caching mechanisms to enhance performance.

Data security and Protection

The Data and Security Manager has a fundamental responsibility to protect data from unauthorized access, breaches, or misuse. They have technical skills in encryption, data obfuscation, and data privacy to ensure that data can only be accessed by those authorized to do so.

Data Privacy and Regulatory Practices

Most data products operate within a single country and regulatory regime, and still others may require operations in multiple jurisdictions. In either case, the Data and Security Manager is aware of best practices to ensure compliance with data protection regulations, such as GDPR or HIPAA, and works closely with enterprise IT and security teams to establish data security protocols.

Data Lifecycle Management

The Data and Security Manager is responsible for defining and implementing processes for data acquisition, storage, retention, archival, and disposal. They have the skills necessary to manage the Data Product's data throughout its lifecycle of data assets to optimize storage costs, ensure compliance, and support data-driven decision-making.

Continuous Learning and Adaptability

Like each Data Product team member, the Data and Security Manager must be aware of the trends in data management. This field in particular is changing rapidly and the Data and Security Manager should have a desire and enthusiasm for continuous learning and adaptability to stay updated with industry trends.

By fulfilling these responsibilities and possessing these skills, a Data and Security Manage ensures that all data managed by the Data Product, as well as any data ingested, and any data consumed by users is done in a safe and secure manner.

Consumption Services Manager

The Consumption Services Manager builds, supports, and evolves the services needed to consume the data managed by

the data product. They have skills in developing interoperable interfaces and providing the necessary tools and services for users to access and utilize the data effectively.

The Consumption Services Manager plays a crucial role in developing and supporting the services that enable consumers to effectively access and utilize the data managed by data products within a data mesh. Their responsibilities revolve around designing, building, and evolving the consumption services that facilitate data consumption and integration. Here are some of the most important responsibilities and skills of a Consumption Services Manager:

Service Design and Development

The Consumption Services Manager is responsible for designing and developing the services that allow consumers to access and consume data from data products. They work closely with data product owners, data engineers, and other stakeholders to understand consumer requirements and design service interfaces that meet their needs. They should have expertise in service-oriented architecture (SOA) and API design principles.

Interoperability and Integration

The Consumption Services Manager ensures that the consumption services are interoperable with various systems, tools, and technologies used by consumers. They should have a strong understanding of data integration techniques and standards, such as RESTful APIs, message queues, or event-driven architectures. They enable seamless integration between data products and consumer systems to support data-driven applications and workflows.

Service Support and Maintenance

Once the consumption services are deployed, the Consumption Services Manager is responsible for providing ongoing support and maintenance. They address any issues or performance concerns related to the consumption services and work closely with consumers to ensure their smooth operation. They should have strong troubleshooting and problem-solving skills to quickly resolve any service-related issues.

Service Performance and Scalability

The Consumption Services Manager monitors and optimizes the performance and scalability of the consumption services. They analyze service usage

patterns, identify potential bottlenecks, and implement optimizations to enhance service performance and ensure scalability as data volumes and consumer demands increase. They should have a good understanding of performance testing and tuning techniques.

Metadata and Documentation

The Consumption Services Manager ensures that appropriate metadata and documentation are available for the consumption services. They document service interfaces, data schemas, and usage guidelines to facilitate consumer understanding and adoption. They work closely with the metadata manager to ensure that the necessary metadata is captured and made available to consumers.

Security and Access Control

The Consumption Services Manager ensures the security and access control of the consumption services. They implement authentication, authorization, and encryption mechanisms to protect data and restrict access based on consumer roles and permissions. They should have a solid understanding of data security principles and best practices to ensure the confidentiality and integrity of data.

Communication and Stakeholder Management

Effective communication and stakeholder management are crucial for the Consumption Services Manager. They work closely with Data Product Owners, users that interact with the Data Product, and other stakeholders to understand their requirements, gather feedback, and ensure that the consumption services meet their needs. They should have excellent interpersonal and communication skills to facilitate collaboration and manage expectations.

Continuous Improvement and Technology Adoption

The Consumption Services Manager should have a mindset of continuous improvement. They actively seek opportunities to enhance the capabilities and efficiency of the consumption services. They stay updated with emerging technologies and trends in service development and integration to leverage new tools, frameworks, or methodologies that can improve the effectiveness and agility of the consumption services.

By fulfilling these responsibilities and possessing these skills, a Consumption Services Manager can enable seamless and efficient access to data products within a data mesh. Their

expertise in service design, integration, performance optimization, and stakeholder management contributes to the successful adoption and utilization of data products by consumers, driving data-driven decision-making and value creation within the organization.

Ingestion Services Manager

The Ingestion Services Manager is responsible for building, supporting, and evolving the services needed to ingest data into the data product. They possess expertise in data engineering, SQL, and pipeline techniques required to efficiently bring data into the data product.

The Ingestion Services Manager plays a crucial role in designing, building, and maintaining the services responsible for ingesting data into data products within a data mesh. Their responsibilities revolve around ensuring the smooth and efficient flow of data from various sources into the data products. Here are some of the most important responsibilities and skills of a data product ingestion services manager:

Data Ingestion Strategy

The Ingestion Services Manager is responsible for defining the overall data ingestion strategy for the Data

Product. They collaborate with Data Product Owners, engineers, developers, and other stakeholders to understand data source requirements, data quality considerations, and data integration needs. They develop an effective strategy that encompasses data acquisition, transformation, and loading processes.

Data Pipeline Design and Development

The Ingestion Services Manager designs and develops data pipelines or workflows that enable the extraction, transformation, and loading of data into the data products. They employ various techniques and tools to facilitate seamless and automated data ingestion processes. They should have expertise in data integration technologies, such as ETL (Extract, Transform, Load) tools, data streaming platforms, or data integration frameworks.

Data Transformation and Validation

Data ingestion often involves data transformation and validation tasks to ensure the quality and consistency of the ingested data. The Ingestion Services Manager is responsible for designing and implementing these transformation and validation processes. They should

have a solid understanding of data manipulation techniques, data cleansing methods, and data validation rules.

Data Source Connectivity

The Ingestion Services Manager establishes connectivity with various data sources, including databases, APIs, file systems, streaming platforms, or other external systems. They work closely with data source owners and administrators to define the integration requirements, establish secure connections, and ensure the efficient extraction of data from the sources. They should have knowledge of data access protocols, connectivity options, and data source-specific considerations.

Performance Optimization

The Ingestion Services Manager focuses on optimizing the performance and efficiency of data ingestion processes. They monitor and analyze the ingestion workflows, identify performance bottlenecks, and implement optimizations to enhance data ingestion speed, scalability, and reliability. They should have a good understanding of performance tuning techniques and data pipeline monitoring tools.

Data Governance and Compliance

The Ingestion Services Manager ensures that data ingestion processes adhere to data governance policies and regulatory compliance requirements. They collaborate with data governance teams and stakeholders to establish data quality standards, data privacy measures, and data classification rules. They should have knowledge of data governance frameworks, data protection regulations, and data privacy best practices.

Error Handling and Exception Management

Ingesting data from various sources can involve challenges such as data discrepancies, errors, or exceptions. The Ingestion Services Manager is responsible for implementing error handling mechanisms and exception management processes. They should have strong troubleshooting and problem-solving skills to identify and resolve data ingestion issues promptly.

Collaboration and Stakeholder Management

Effective collaboration and stakeholder management are essential for the Ingestion Services Manager. They work closely with data product owners, data engineers, data source owners, and other stakeholders to understand

their requirements, gather feedback, and ensure that the data ingestion services meet their needs. They should have excellent communication and interpersonal skills to facilitate collaboration and manage expectations.

Continuous Learning and Adaptability

Like the other roles, the Ingestion Services Manager must take into account the constantly changing data management landscape. The Ingestion Services Manager should have a passion for continuous learning and adaptability to stay updated with industry trends and incorporate new techniques into their data ingestion processes. They should be open to exploring innovative approaches and leveraging new tools and technologies to improve data ingestion efficiency.

By fulfilling these responsibilities and possessing these skills, a Ingestion Services Manager can ensure the smooth, efficient, and reliable ingestion of data into data products within a data mesh. Their expertise in data integration, pipeline design, performance optimization, and stakeholder management contributes to the successful acquisition and preparation of data for consumption, enabling data-driven decision-making and value creation within the organization.

Data Product Skills Matrix

Now that we understand the various teams, and in particular the roles and responsibilities of the Data Product team, let's take a look at the actual skills needed to populate the team.

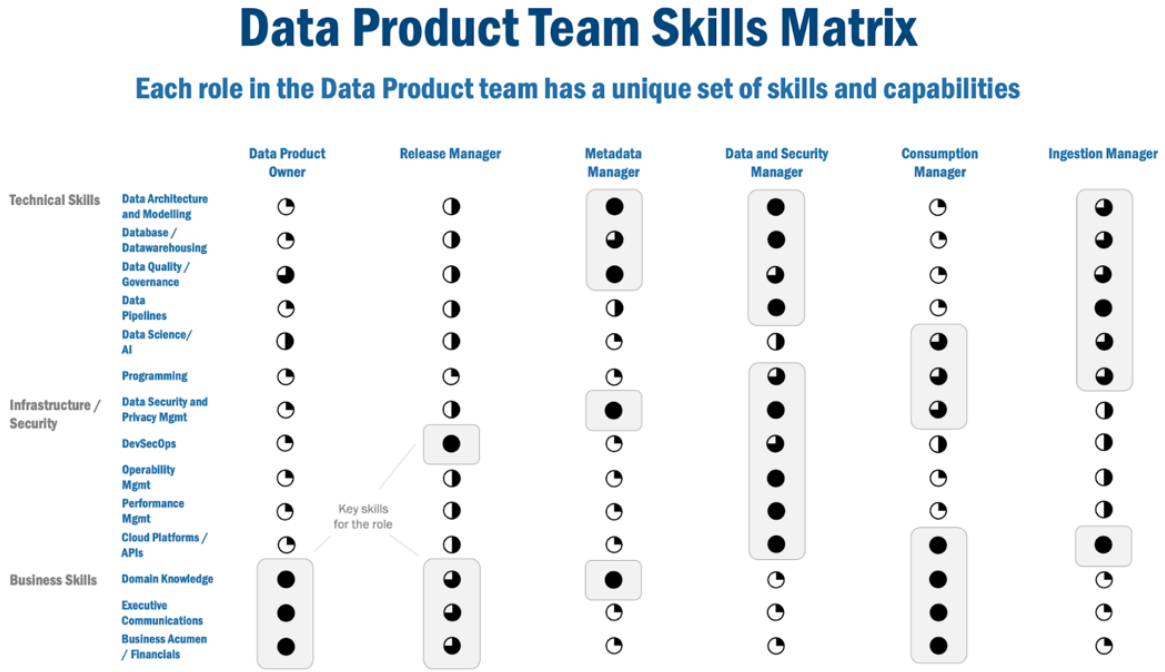


Figure 14-3. Data Product Team Skills Matrix

The above figure provides a skills matrix for the Data Product team, but there are some clear skills that are emphasized for each team role:

Data Product Owner

As the owner of the end-to-end Data Product capabilities, this role emphasizes the business skills. The Data Product Owner must have strong domain knowledge and speak in the language of the business community the Data Product serves. As the spokesperson for the Data Product team, this role must also have outstanding communication and sales skills. And with business accountability for the execution and operations of the Data Product, this role must also have a strong knowledge of finance to ensure funding is available to meet expectations.

Release Manager

The ideal Release Manager is a “jack of all trades” requiring a reasonable understanding of the technical domain, in particular DevSecOps, to ensure new versions and releases of Data Product capabilities are delivered successfully. But this role is also forefront in communicating with the broader data product consumer and provider community and hence the Release Manager must also have a strong domain knowledge as well as communication skills.

Metadata Manager

This is a role that focuses on governance, quality, and understanding of the data within the Data Product. This role must be fluent with regulatory and organizational practices of the enterprise and hence must have strong business domain knowledge. But they also must have strong technical skills, but in a specialized area around management of “metadata” - expertise in data catalogs and data contracts are crucial in this role.

Data and Security Manager

This role secures and manages the data within the Data Product, and hence requires an outstanding knowledge of the data platform technology (database, datalake, datawarehouse etc). With security and privacy playing a crucial role in all modern enterprises, this role is an expert in practices that ensure access to the data product and its data is only available to authorized individuals, that all data is secure and protected.

Consumption Manager

This role ensures that the data product is accessible and available to consumers of the data product. As such this role has deep technology data access skills such as programming and APIs, as well as in data science and

analytics which would be common usage for the Data Product.

Ingestion Manager

This role ensures that data is ingested and stored in the data product in a secure and reliable manner. As such, this role has deep technical skills in data pipelines and data transformation, as well as data storage skills related to the data platform(database, datalake, datawarehouse) used by the data product.

Benefits

Data Product teams bring several benefits to organizations when it comes to effectively managing and leveraging data. Here are some key benefits of having a Data Product team:

Data-Driven Decision Making

Data Product teams enable organizations to make data-driven decisions by providing reliable and actionable insights. They work closely with stakeholders to understand their data needs and develop products and services that meet those needs. By leveraging data analytics and visualization techniques, data product

teams empower decision-makers with accurate and timely information.

Ownership and Accountability

Data Product teams take ownership of specific data products or services, ensuring end-to-end accountability for their success. This ownership mentality promotes responsibility, proactive problem-solving, and continuous improvement. Having dedicated teams responsible for data products helps ensure that they are well-maintained, updated, and aligned with business goals.

Faster Time-to-Value

Data Product teams are focused on delivering value through efficient data management, processing, and analysis. By streamlining the data pipeline and employing agile methodologies, these teams can accelerate time-to-value by reducing bottlenecks, improving data accessibility, and quickly responding to changing business needs.

Domain Expertise

Data Product teams develop deep domain expertise in the specific areas they work on. They understand the

intricacies and nuances of the data, as well as the unique challenges and opportunities within the domain. This expertise enables them to provide valuable insights and solutions tailored to the specific needs of the business.

Collaboration and Alignment

Data Product teams collaborate with various stakeholders, including data producers, data consumers, and other teams within the organization. This collaboration ensures that the data products align with business objectives and meet the requirements of users. By working closely with stakeholders, Data Product teams can bridge the gap between technical capabilities and business needs.

Scalability and Flexibility

With a dedicated team focused on Data Products, organizations can scale their data initiatives more effectively. Data Product teams can adapt to changing data requirements, accommodate growth, and address evolving challenges. They have the flexibility to leverage new technologies and approaches as the data landscape evolves.

Continuous Improvement

Data Product teams promote a culture of continuous improvement. They regularly gather feedback, analyze metrics, and iterate on data products to enhance their performance and value. This iterative approach helps refine Data Products over time and ensures they remain relevant and effective.

Challenges

Nevertheless, Data Product teams can face several challenges throughout their journey. Here are some common challenges that data product teams may encounter that will likely require constant care and feeding:

Data Quality and Governance

Ensuring data quality and establishing proper data governance processes can be a significant challenge. Data Product teams need to navigate issues related to data accuracy, consistency, completeness, and privacy. Implementing effective data governance frameworks and ensuring data quality standards are maintained are ongoing challenges.

Collaboration and Communication

Data Product teams often need to collaborate with various stakeholders, including producer teams, consumer teams, platform teams, and enabling teams. Ensuring effective communication, coordination, and alignment among these teams can be challenging, particularly when they have different priorities, workflows, or technical requirements.

Evolving Technology Landscape

The rapid pace of technological advancements in the data space can present challenges for Data Product teams. Keeping up with new tools, technologies, and best practices requires continuous learning and adaptation. Additionally, integrating new technologies into existing systems and workflows can be complex and time-consuming.

Scalability and Performance

As Data Products grow and more users rely on them, scalability and performance become crucial challenges. Managing increasing data volumes, handling data processing and storage efficiently, and ensuring responsiveness and reliability can be demanding tasks for data product teams.

Balancing Innovation and Stability

Data Product teams often need to balance the need for innovation and introducing new features with the stability and reliability of existing data products. Striking the right balance between pushing boundaries and maintaining robustness can be a delicate challenge.

Stakeholder Expectations and Alignment

Data Product teams must manage expectations and align with various stakeholders, including business users, executives, and regulatory bodies. Understanding their needs, translating them into actionable requirements, and delivering value that meets their expectations can be challenging, especially when priorities and requirements evolve.

Talent and Skills Gap

Finding and retaining skilled professionals with expertise in data management, data engineering, data analysis, and other relevant areas can be a challenge. The demand for data-related skills often exceeds the available talent pool, leading to talent shortages and skill gaps within Data Product teams.

Summary

In this chapter, we explored the concept of establishing an effective data product team within a data mesh. We discussed the organizational structure and roles within a data mesh, including stream aligned teams, platform teams, enabling teams, and data product teams. A data product team is responsible for the end-to-end delivery of services required by a data product and interacts with producer teams, consumer teams, platform teams, enabling teams, and complicated subsystem teams.

We also delved into the responsibilities and skills of key roles within a data product team, such as the data product team owner, release manager, metadata manager, consumption services manager, and ingestion services manager. These roles encompass a range of responsibilities, including strategic planning, stakeholder management, data governance, data pipeline design and development, performance optimization, and compliance.

Additionally, we discussed the challenges faced by data product teams, the benefits of having a data product team structure, and the requirements for setting up a data mesh within an organization. Overall, establishing a data mesh requires careful

consideration of both technical and organizational aspects, and effective data product teams play a vital role in the success of data mesh implementation, enabling the organization to leverage data-driven insights and value creation.

Chapter 15. Defining an Operating model for Data Mesh

Introduction

A NOTE FOR EARLY RELEASE READERS

With Early Release ebooks, you get books in their earliest form—the author's raw and unedited content as they write—so you can take advantage of these technologies long before the official release of these titles.

This will be the 15th chapter of the final book. Please note that the GitHub repo will be made active later on.

If you have comments about how we might improve the content and/or examples in this book, or if you notice missing material within this chapter, please reach out to the editor at sevans@gmail.com.

An operating model serves as a blueprint for how an organization functions, detailing the interactions and processes that enable it to achieve its goals. This model encompasses aspects such as organizational structure, technology, and

workflows, providing a comprehensive framework for executing business strategies.

The translation of this concept to the realm of data management, particularly in the context of an enterprise Data Mesh, offers a novel approach to handling complex data ecosystems. A Data Mesh, advocating for a decentralized approach to data management, treats data as a product and emphasizes domain-oriented ownership. Therefore, crafting an operating model for a Data Mesh involves rethinking traditional centralized data management paradigms and embracing a more distributed, agile, and domain-centric approach.

The transition to a Data Mesh operating model necessitates a shift in organizational behavior, which often proves more challenging than integrating new technologies. This change involves establishing new roles, such as data product owners, and redefining interactions between various teams within the data ecosystem. The operating model must support these roles and interactions, fostering collaboration and efficiency.

The journey towards an effective Data Mesh operating model is about aligning the organization's structure, culture, and technology with the principles of decentralized data management, empowering teams to manage and utilize data

more effectively, driving innovation and efficiency. The subsequent sections will delve into the specifics of this model, exploring the types of teams involved, their interactions, and the broader implications of this approach on the enterprise's data strategy.

There are several key aspects of the Data Mesh operating model that will be elaborated upon:

- Operating Model Characteristics: We will highlight different operating model types and their characteristics across a continuum from centralized to fully distributed
- Data Mesh Ecosystem Operating Model: In this section, we discuss the data product team operating model and how it integrates with a broader data mesh operating model
- Data Governance vs Data Certification: Since the policies and enforcement of policies is a crucial aspect of a data mesh operating model, we explore how data governance can be simplified
- Operating Model Implications: Conway's Law - that systems and data will mirror your organization structure - has material impact on your operating models. This section will discuss its implications.
- Data Mesh as Loosely Coupled Regional Ecosystems: We will discuss the trend towards regional data mesh

implementations due to organizational, regulatory, and technological factors, and how these regional ecosystems interact within the larger enterprise.

Characteristics of an Operating Model

An operating model is a blueprint that defines how an organization conducts business, delivering value through its operations. It aligns people, processes, and technology to achieve strategic goals. In the context of data products or a Data Mesh, the operating model guides how data is managed, shared, and utilized within an organization, ensuring that data initiatives align with business objectives and strategies.

One key objective of an operating model is to establish clear roles and responsibilities, ensuring that each team and individual knows their specific functions and how they contribute to the broader organizational goals. In a Data Mesh, this means defining the roles of various teams, such as Data Product teams, Platform teams, and Enabling teams. For example, Data Product teams manage and curate specific data sets, Platform teams provide the necessary infrastructure and tools, and Enabling teams offer support and expertise. This

clear delineation of roles ensures efficient management and utilization of data across the organization.

Another important characteristic of an operating model is the establishment of efficient processes. These processes dictate how tasks and activities are carried out within the organization, optimizing performance and output. In the realm of data products, this means establishing standardized processes for data governance, quality control, and lifecycle management. These processes ensure that data products are reliable, up-to-date, and compliant with relevant regulations and standards, thereby enhancing their value and usability within the Data Mesh framework.

The integration of technology is also a crucial aspect of an operating model. It involves selecting and implementing the right technological tools and platforms that support the organization's operations. In a Data Mesh, this aspect is particularly significant as it involves choosing the right data storage, processing, and analytics tools that empower Data Product teams to effectively create and manage their data products. The operating model should facilitate a technology landscape that is agile, scalable, and conducive to the decentralized nature of a Data Mesh, enabling seamless interaction and data sharing across different domains.

Furthermore, an operating model emphasizes the importance of communication and collaboration among different units within an organization. Effective communication channels and collaboration practices ensure that teams can work together seamlessly, sharing insights and expertise. In a Data Mesh environment, this is vital for the success of data products, as it involves constant interaction between Data Product teams, stakeholders, and consumers of the data. Collaboration fosters a culture of shared responsibility and collective effort in data management, leading to more innovative and effective use of data across the enterprise.

Lastly, adaptability and continuous improvement are inherent characteristics of a robust operating model. It should be flexible enough to evolve with changing business needs and technological advancements. In the context of data products and a Data Mesh, this means the operating model must accommodate emerging data technologies, changing data privacy regulations, and evolving business strategies. It should encourage a culture of learning and adaptation, where feedback and insights lead to continuous refinement of data practices and strategies. This adaptability ensures that the organization remains agile and responsive, keeping its data capabilities relevant and effective in a dynamic business environment.

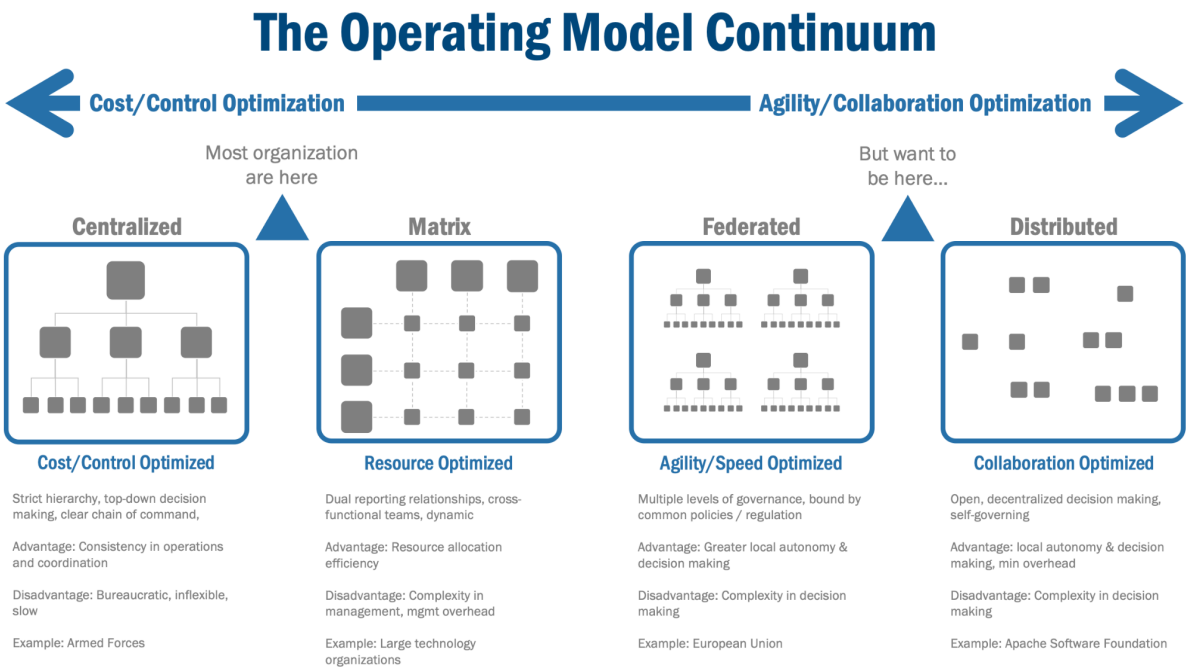


Figure 15-1. The Operating Model Continuum

There are several destinations on the operating model continuum, as shown in Figure 15-1.

Centralized organizations, exemplified by a nation's armed forces, are characterized by their emphasis on cost optimization and control. They often employ a strict hierarchy with top-down decision-making which establishes a clear chain of command, where decisions and directives flow from the top level of the organization down to the lower levels. Such a hierarchy ensures that all parts of the organization are aligned with the central vision and policies, which is critical for maintaining uniformity in practices and standards. In industries where consistency and uniformity are essential, like

manufacturing or finance, this centralized approach can be particularly effective. The predictability and control afforded by a centralized model are conducive to streamlined processes and clear accountability, ensuring that the entire organization moves cohesively towards its objectives.

However, the advantages of a centralized organizational model are accompanied by certain disadvantages, particularly related to its rigid structure. The top-down approach to decision-making can lead to a bureaucratic and inflexible environment. In situations where quick, responsive decision-making is required, this rigidity can be a significant handicap. The lack of local autonomy means that decisions are made by those at the top, often detached from the on-ground realities and frontline insights. This can result in decisions that are not well-suited to local or departmental needs, leading to inefficiencies and frustration among employees. Moreover, the centralization of decision-making can create bottlenecks, as all major decisions must go through a few high-level executives, slowing down the organization's ability to respond to changes in the market or internal challenges.

Additionally, centralized organizations might struggle with innovation and adaptability. In a fast-paced, constantly evolving business landscape, the ability to innovate and adapt quickly is

crucial. A centralized structure, with its lengthy decision-making processes and hierarchical rigidity, may impede the flow of new ideas and inhibit the organization's ability to pivot swiftly in response to new opportunities or threats. Employees in such organizations may feel less empowered to suggest changes or innovations, as the decision-making power is concentrated at the top. This can lead to a lack of motivation and engagement among the workforce, who may feel their insights and local knowledge are undervalued. In essence, while centralized organizations benefit from uniformity and control, they must navigate the challenges of inflexibility, potential bureaucratic delays, and a possible dampening of innovative spirit.

Matrixed organizations, on the the hand, feature dual reporting relationships and cross-functional teams, which contributes to a dynamic and collaborative work environment. In this model, employees typically report to both a functional manager and a project manager, integrating expertise from various disciplines into one team. For instance, an employee in a technology company might report to an IT department manager (functional manager) and also to a project manager leading a specific initiative, like developing a new software product. This dual reporting structure is intended to optimize the utilization of skills and expertise, ensuring that project teams have access to

a diverse range of talents and perspectives. Cross-functional teams in a matrixed setup bring together specialists from different areas, such as engineering, marketing, and finance, fostering a holistic approach to project development and problem-solving.

One significant advantage of matrixed organizations is the efficiency in resource allocation. The flexibility of the matrix structure allows for the optimal deployment of human resources across various projects, as employees can be assigned to tasks that best fit their skills and experience, irrespective of their departmental affiliation. This not only maximizes the use of the workforce but also contributes to a more agile and adaptable organization. In large technology organizations, where project scopes and resource needs can fluctuate rapidly, this ability to swiftly reallocate resources is particularly beneficial. It ensures that projects are not delayed due to resource shortages and that employees are consistently engaged in meaningful and challenging work, contributing to higher levels of job satisfaction and productivity.

However, matrixed organizations also face notable challenges, particularly in terms of management complexity and increased managerial overhead. The dual reporting lines can lead to confusion and conflict, as employees may receive competing

directives or priorities from their functional and project managers. This complexity requires effective communication and clear role definitions to avoid ambiguities and conflicts. Additionally, the need to coordinate between multiple managers and teams can lead to increased management overhead, with more time spent on meetings and communication to align on objectives and strategies. In large technology companies, where projects are often complex and involve many stakeholders, this can result in slower decision-making processes and potential inefficiencies. Managers in a matrixed organization need to possess strong leadership and conflict-resolution skills to navigate these complexities effectively and maintain a productive work environment.

Now, let's take a look at federated organizations. They are typically structured in a way that combines multiple levels of governance, each with a degree of autonomy, yet all bound together by a set of common policies and regulations. This structure is particularly evident in complex entities like the European Union (EU), where each member state retains its sovereignty while agreeing to adhere to certain overarching policies and regulations set by the EU. The governance within a federated organization is tiered, with some decisions and policies being made at the highest level, and others at more localized levels. This multi-tiered approach allows for a balance

between unity and diversity, enabling different parts of the organization to function cohesively towards shared goals while respecting their individual characteristics and needs.

One of the primary advantages of federated organizations is the greater degree of local autonomy and decision-making power they offer. In the context of the EU, member states have the freedom to make decisions on many internal matters, allowing them to address local needs and preferences effectively. This autonomy is crucial in ensuring that the diverse cultural, economic, and political contexts of each member are taken into account. Local autonomy fosters a sense of ownership and responsibility among the members, as they are not merely following directives from a central authority but are actively involved in the governance process. This can lead to more effective and tailored policy implementations, as decisions are made by those who are closely acquainted with the specific contexts and challenges.

However, the federated structure also brings with it a certain complexity in decision-making. The need to align and coordinate policies across different levels of governance can lead to lengthy negotiations and compromises. In the EU, for instance, reaching consensus among all member states can be a challenging and time-consuming process, especially on

contentious issues where national interests may diverge. This complexity can sometimes slow down the decision-making process, leading to delays in policy implementation. Furthermore, the need to accommodate diverse viewpoints can result in policies that are less effective or diluted in their impact. The challenge for federated organizations lies in finding the right balance between respecting local autonomy and ensuring efficient and effective decision-making at the broader level.

Lastly, distributed organizations are characterized by their open, decentralized decision-making processes and a self-governing approach. This model is exemplified by entities like the Apache Software Foundation (ASF), where projects are managed independently by various teams dispersed across different locations. In a distributed organization, decision-making authority is not centralized in a single management hierarchy but is spread across multiple nodes or units within the organization. Each unit operates with a high degree of autonomy, making decisions that are best suited to their specific context and objectives. This structure is often enabled and supported by digital communication technologies that facilitate coordination and collaboration among geographically dispersed teams.

One of the key advantages of distributed organizations is the high degree of local autonomy and decision-making power they provide. This autonomy allows for decisions to be made closer to where they will have their impact, leading to more responsive and contextually appropriate outcomes. For example, in the ASF, various project teams are empowered to make decisions regarding their specific software projects, leading to more efficient and innovative development processes. Additionally, distributed organizations tend to have minimal bureaucratic overhead. Without the need for a large central administrative structure, these organizations can operate more leanly, reducing costs and increasing operational efficiency. The decentralized nature also encourages a more entrepreneurial and innovative environment, as teams are not constrained by a rigid central policy.

However, the distributed model also comes with its own set of challenges, particularly in terms of decision-making complexity. The lack of a centralized decision-making authority can sometimes lead to inconsistencies and difficulties in maintaining a unified strategic direction. In the case of the ASF, coordinating efforts and maintaining consistency across numerous independent projects can be a complex task. The challenge lies in ensuring that all the autonomous units or teams are aligned with the overall objectives and values of the

organization while retaining their independence. Additionally, in the absence of a central authority, conflict resolution and the enforcement of standards and policies can become more complicated. The success of a distributed organization like the ASF depends heavily on the establishment of strong, shared values and objectives, along with effective communication channels and collaborative tools that enable disparate teams to work towards common goals while respecting each other's autonomy.

Now, at the end of the day, each organization is unique and probably has a mix of several operating models, each implemented in different parts of the enterprise. Nevertheless, it is important to understand the advantages and disadvantages of each operating model to be able to plan your approach for your data mesh and data products.

Data Mesh Ecosystem Operating Model

Independent of your actual operating model, and how data products manifest themselves in your organization, one thing is absolutely clear: data products, as the quantum unit within a data mesh, are built and maintained by teams.

So, let's explore the makeup of the data product team and then apply it to your operating model.

The Data Product Team Operating Model in a Data Mesh framework defines how data product teams function within the broader organizational context. This model is pivotal for ensuring that Data Product teams are effective, autonomous, and aligned with the overall objectives of the Data Mesh. The operating model serves as a microcosm of the organization's broader operating model, tailored to the specific needs and objectives of data management.

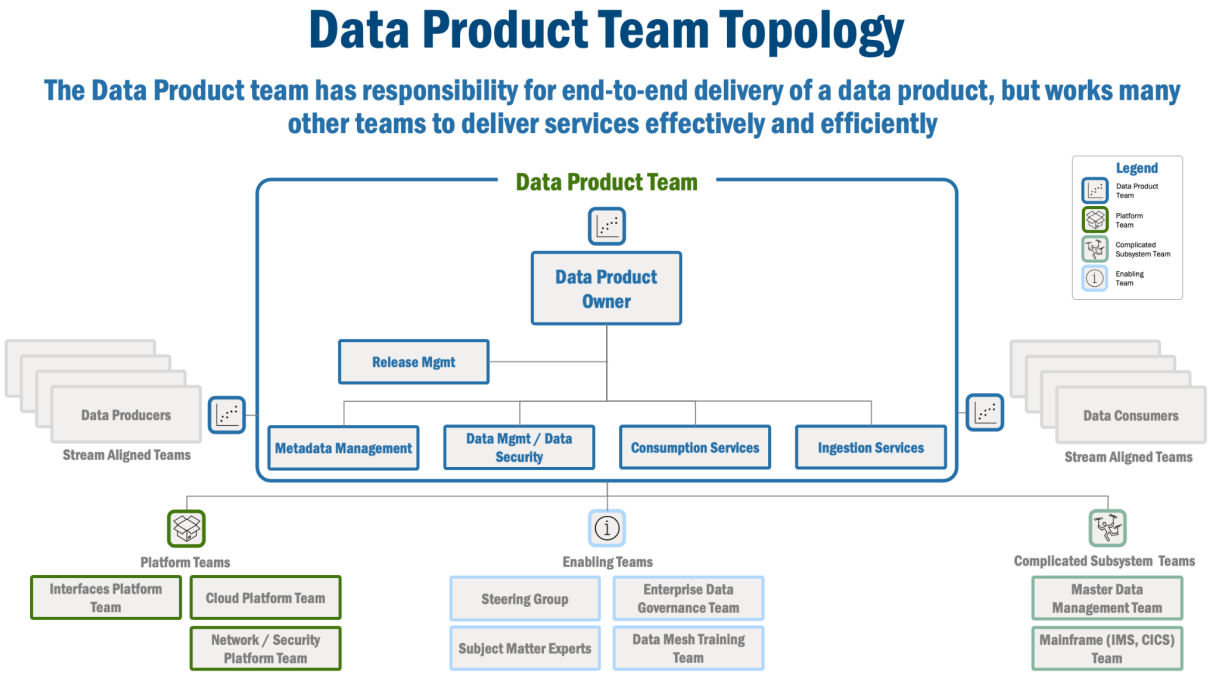


Figure 15-2. The Operating Model Continuum

The primary objective of the Data Product Team, an example of which is shown in Figure 15-2, is to establish clear roles and responsibilities within these teams. This clarity helps in delineating the specific functions each team member performs, from data curation and quality assurance to analytics and reporting. The outcome is a team that operates efficiently, with each member contributing their expertise towards the creation and maintenance of high-quality data products. This model also emphasizes the importance of autonomy. Data Product teams are given the authority to make decisions regarding their data products, from how data is collected and processed to how it's distributed and used. This autonomy fosters a sense of ownership and responsibility, driving teams to be more innovative and responsive to changing data needs. Note that the specific responsibilities of each team is described in a prior chapter.

In our Data Mesh Ecosystem Operating Model, we extend its scope beyond the realm of individual Data Product teams, encapsulating the entire spectrum of data management within an organization. This model is pivotal in orchestrating the interactions and collaborative efforts of multiple data product teams, aligning them to form an integrated and efficient Data Mesh. This holistic approach is crucial in harnessing the full potential of a decentralized data architecture, ensuring that the

collective output of various data products synergizes into a unified, strategic asset for the organization.

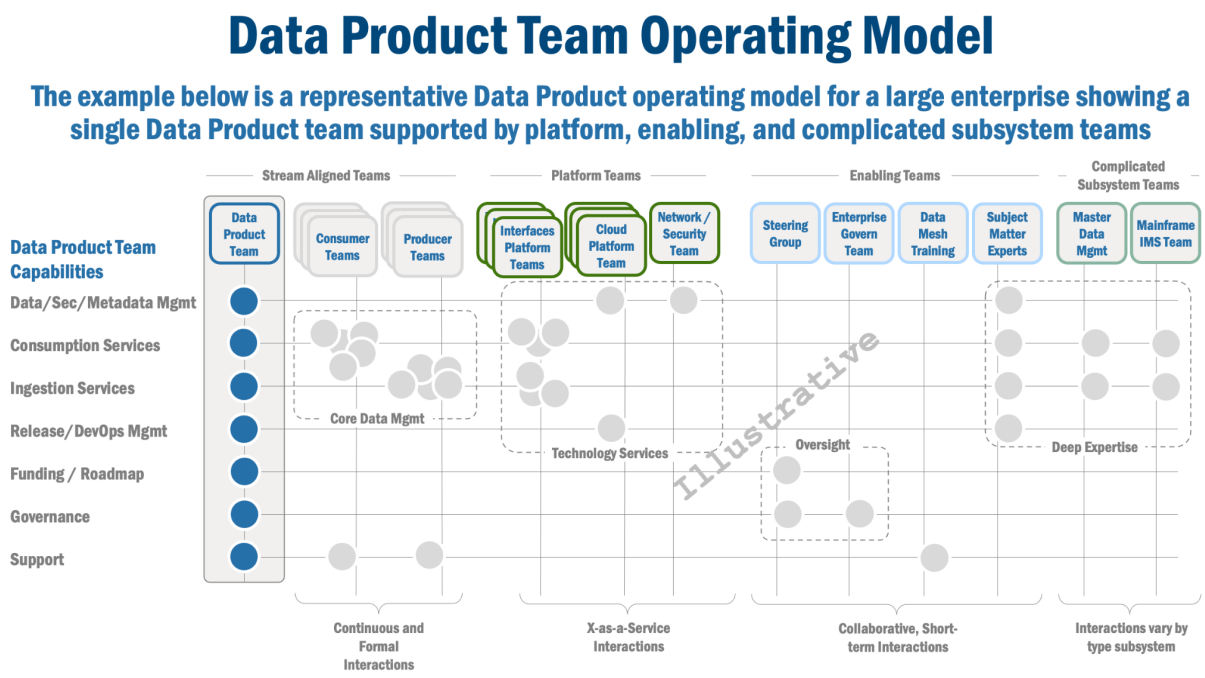


Figure 15-3. Operating Model Interactions

Central to this model is the integration of the individual Data Product team operating models into a larger, interconnected ecosystem. While these teams concentrate on their domain-specific roles, the Ecosystem Operating Model ensures they do not operate in isolation. It promotes a culture of collaboration and shared learning, encouraging teams to exchange best practices, tools, and insights. This synergy is essential for balancing the autonomy of individual teams with the overarching goals of the Data Mesh, facilitating data products

that are not only effective in their own right but also interoperable and complementary across the organization.

One of the main objectives of this model is to foster a seamless flow of data and collaboration among different Data Product teams. This goal is achieved by creating an environment conducive to easy sharing of data, knowledge, and resources, thus amplifying the collective value of the Data Mesh. Such an environment paves the way for a more agile and responsive data infrastructure, adaptable to the dynamic needs of business and technology landscapes. Another critical objective is maintaining consistency and uniformity in data practices and standards across the Mesh, which is instrumental in upholding data quality and reliability.

Data Certification vs Traditional Data Governance

Incentives within the broader Data Mesh ecosystem are aligned to promote collaboration and innovation while adhering to shared standards. Recognition programs for teams that effectively integrate their data products or contribute to the Data Mesh's overall efficiency are examples of such incentives. These rewards foster a culture where teams are driven not only

to excel in their specific domains but also to contribute meaningfully to the collective data strategy.

Governance within this broader ecosystem follows a federated, certification-based model, akin to the American National Standards Institute (ANSI) approach. This model stipulates overarching policies and standards at the ecosystem level, while allowing Data Product teams to retain their autonomy. Teams align their practices with these broader standards to ensure cohesion and consistency throughout the Data Mesh. Such a federated governance structure strikes a balance between local independence and strategic alignment.

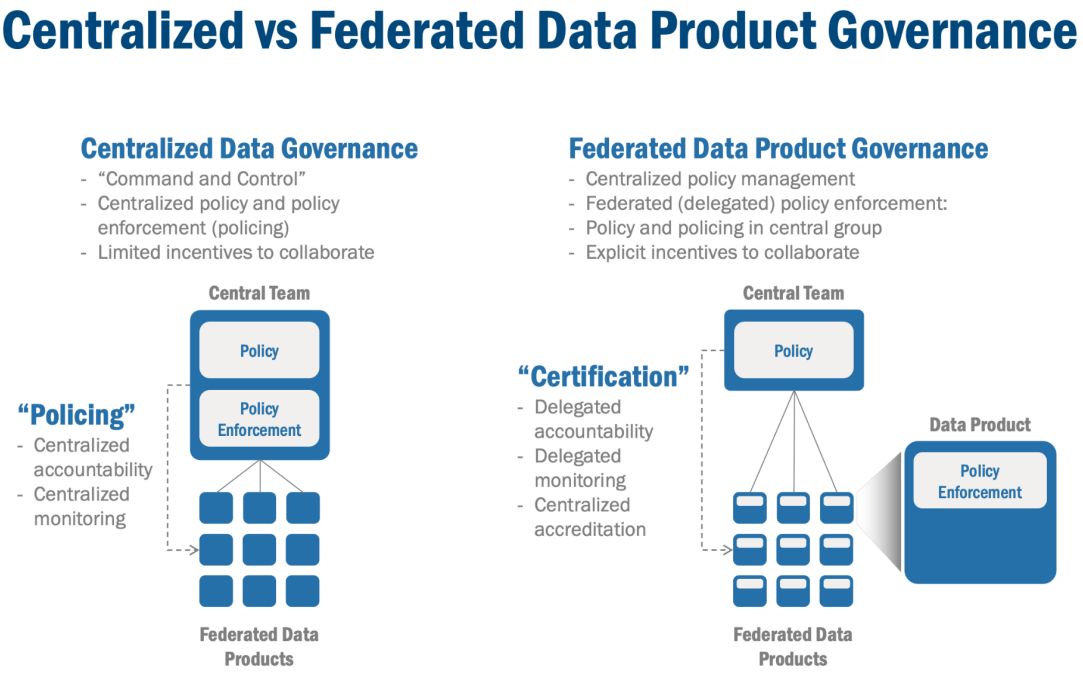


Figure 15-4. Centralized vs Federated Data Product Governance

A central, but light-weight and nimble, governance entity, mirroring the role of the ANSI, is integral to this model, shown in Figure 15-4. This body sets forth the core policies, standards, and certification processes for the entire Data Mesh. Rather than enforcing compliance top-down, it establishes a framework within which all Data Product teams operate, ensuring that these standards stay relevant, current, and aligned with both organizational objectives and industry best practices. By doing so, it upholds a high standard of data quality, security, and compliance across the Mesh while fostering the flexibility needed for teams to innovate and address their unique data challenges.

Governance within the Data Product Team Operating Model aligns more with a “certification” approach rather than a traditional centralized policing style. A central governance team establishes policies and standards, and it is up to the individual entities, in this case, the Data Product teams, to adhere to these standards. The teams are responsible for ensuring their data products meet the established criteria, much like a vendor ensuring their product meets certain quality standards before it gets certified.

This governance model empowers Data Product teams to take charge of their compliance, encouraging them to integrate

governance practices into their daily operations. The certification approach also fosters transparency, as teams publish their compliance or certification status, making it clear to stakeholders and consumers of the data products how they adhere to set standards. This transparency builds trust within the organization and with external partners, as it provides assurance that data products are managed responsibly and in alignment with organizational and regulatory requirements.

The certification approach to governance benefits the Data Mesh in several ways. Firstly, it reduces the bottleneck often associated with centralized governance models, where a central team has to oversee and approve numerous aspects of data management. By decentralizing governance, Data Product teams can move faster, adapting to new requirements or changes in the data landscape without being held up by lengthy approval processes. Secondly, this approach encourages a culture of continuous improvement. Teams are not just aiming to meet minimum standards; they are incentivized to exceed them, knowing that their efforts will be recognized and rewarded.

For the certification approach to be effective, it is essential that the standards set by the central governance body, like the ANSI analogy, are clear, achievable, and relevant. These standards

should be regularly reviewed and updated to reflect the evolving data landscape and organizational needs. Furthermore, the process for certifying compliance should be straightforward and transparent, with clear guidelines and support for teams navigating the certification process.

Clearly, the Data Product Team Operating Model in a Data Mesh is an approach that combines autonomy, clear roles and responsibilities, and a certification-style governance model which ensures that Data Product teams are empowered to manage their data products effectively while adhering to organizational standards and fostering a culture of innovation, quality, and continuous improvement. And by aligning the operating model with these principles, organizations can leverage the full potential of their data assets within the Data Mesh framework.

Impact of Operating Model Choices

The selection of an operating model for a Data Mesh is a strategic decision with far-reaching implications. It dictates how effectively an organization can utilize its data assets and adapt to evolving business landscapes. This decision-making process involves evaluating trade-offs among key factors such

as agility, centralization, cost efficiency, and control. A nuanced understanding of these trade-offs is essential for crafting an operating model that not only meets current organizational needs but also lays a foundation for future growth and adaptability.

Some of these tradeoffs are described quite well through Conway's law. So, let's start there.

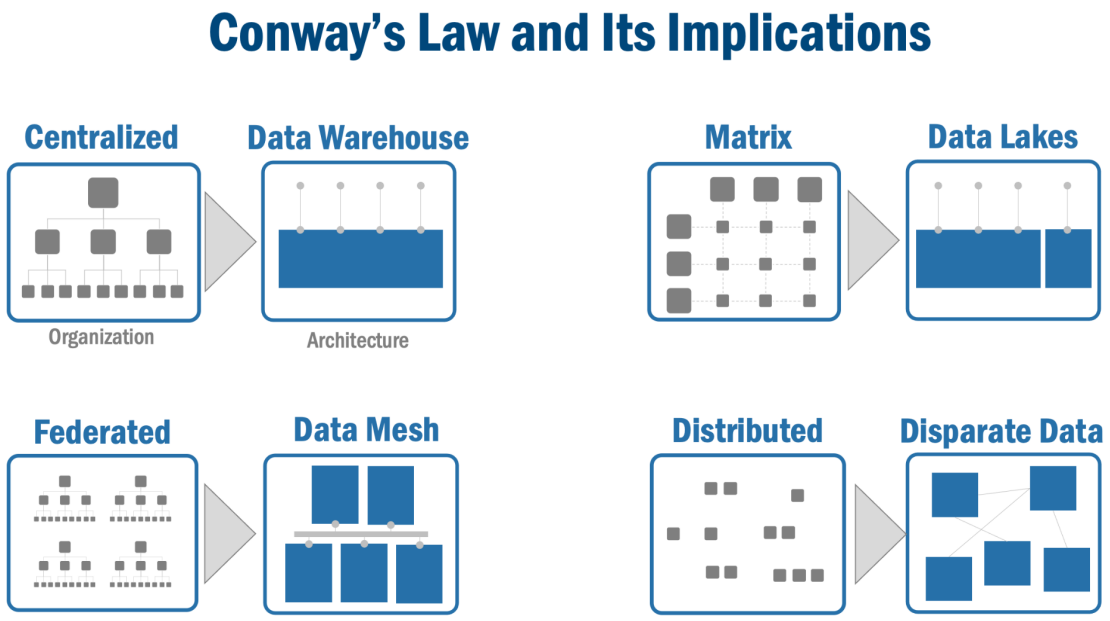


Figure 15-5. Conway's Law and Its Implications

The principles of Conway's Law, which, to paraphrase, states that an organization's systems mirror their organization and communication structures. And the relationship between an organization's operating model and its data architecture is a

critical one. Understanding this relationship is key to implementing successful data management strategies such as data mesh, especially in the context of various organizational structures like centralized, matrixed, federated, and distributed models. So, clearly Conway's law will likely influence the evolution of your Data Mesh.

But let's explore each of the operating models and their implications on architecture, as shown in Figure 15-5.

In centralized organizations, decision-making and control are highly concentrated at the top of the hierarchy. This model is often found in sectors where uniformity and precision are paramount, such as manufacturing or finance. The centralized nature of these organizations lends itself well to monolithic data architectures, like centralized data warehouses. The key advantage of this architecture is its ability to maintain consistency and control over data. A centralized data warehouse ensures that all organizational data adheres to a uniform set of standards and policies, mirroring the centralized decision-making process. However, the rigidity of this model can be a disadvantage in rapidly changing environments, as it may not adapt quickly to new data sources or analytics needs.

On the other hand, matrix organizations, characterized by cross-functional teams and dual reporting structures, require a more flexible data architecture. In these organizations, employees often work on a variety of projects, necessitating access to a wide range of data from different departments. Data lakes are well-suited to this model. A data lake allows for the storage of vast amounts of raw data in its native format, supporting the diverse and dynamic nature of matrix organizations. The flexibility of a data lake aligns with the need for cross-functional teams to access and analyze data from multiple sources. However, the challenge lies in maintaining data quality and governance, as the lack of structure in data lakes can lead to issues like data silos and inconsistent data formats.

In contrast, federated organizations are structured to balance centralized governance with local autonomy, as seen in entities like the European Union. This model is ideal for implementing a data mesh architecture. A data mesh recognizes the distributed nature of data ownership and provides a decentralized approach to data management. It allows different departments or teams within an organization to own and manage their data as individual products, while still adhering to overarching governance standards. This approach supports the federated model by empowering local units with autonomy over their

data, yet ensuring alignment with broader organizational objectives. The challenge with data mesh in a federated organization is ensuring that the decentralized nature of data ownership does not lead to inconsistencies or conflicts in data standards and policies.

Lastly, in distributed organizations, decision-making is highly decentralized, with various units operating independently. This structure is reflected in a microservices architecture for data. Microservices architecture involves breaking down applications into smaller, independent components, each handling a specific function. This approach is beneficial for distributed organizations like the Apache Software Foundation, where different teams work autonomously on various projects. Microservices allow for greater flexibility and scalability, as each service can be developed, deployed, and scaled independently. However, the challenge lies in ensuring effective communication and data integration between these services, as the decentralized nature of both the organizational model and the architecture can lead to difficulties in maintaining a cohesive strategy.

The choice of data architecture is deeply influenced by an organization's operating model. Centralized models favor control and consistency, making centralized data warehouses a

natural choice. Matrix organizations, with their need for flexibility and diverse data access, align well with data lakes. Federated models, balancing centralized control with local autonomy, are conducive to data mesh architectures. Distributed models, characterized by high autonomy, align with microservices architecture. Understanding these linkages is crucial for organizations to design data architectures that support their operational needs and strategic objectives.

Data mesh principles are particularly well-suited to federated operating models due to their emphasis on decentralized data ownership while maintaining a unified governance framework. In a federated model, the autonomy granted to various units aligns with the data mesh principle of domain-oriented decentralized data ownership and architecture. Each unit in a federated organization can manage its data as a product, making decisions that best serve their local context, yet adhering to the broader organizational standards and principles. This alignment supports both the operational flexibility of the units and the strategic coherence of the organization as a whole.

In environments where data products are managed by domain-specific teams, the underlying organizational structure significantly influences the development and maintenance of

these products. For instance, in organizations with a regionalized structure, data products tend to align with regional demands and constraints. This alignment can be a double-edged sword: on one hand, it ensures local relevance and responsiveness; on the other, it may challenge the establishment of a coherent, global data strategy.

The regional focus fostered by Conway's Law often results in data products that are finely tuned to local market needs and regulatory landscapes. While this approach enhances the effectiveness of data products in specific regions, it can inadvertently lead to fragmentation in global data strategy, manifesting as data silos and challenges in cross-regional collaboration. Organizations striving for a unified global data perspective might struggle to integrate these diverse, regionally-focused data products into a harmonious whole.

The dilemma between adopting centralized platforms versus decentralized data products is central to shaping a Data Mesh's operating model. Centralized platforms, characterized by economies of scale and standardization, offer cost-effective solutions with simplified maintenance and enhanced security. However, they may lack the flexibility required to meet the unique demands of diverse data products, potentially curtailing innovation and adaptability.

In contrast, decentralized data products prioritize agility and customization, enabling teams to swiftly respond to specific domain requirements and local market trends. This approach is particularly advantageous in a Data Mesh, where the timely and relevant development of data products is paramount. The downside, however, is the risk of increased costs due to potential duplication of resources and infrastructure. Additionally, without strategic oversight, decentralization might lead to inconsistent standards and practices across the organization's data landscape.

In a decentralized Data Mesh, these strategic choices have profound implications. While decentralization fosters domain-specific innovation and quick adaptation, it necessitates robust governance to ensure alignment with the organization's overarching goals. The key challenge lies in striking an optimal balance: empowering teams to be innovative and agile in their respective domains while maintaining a cohesive standardization that aligns with the organization's broader data strategy. This balance is essential for leveraging the full potential of a Data Mesh, facilitating an organization's ability to effectively harness its data assets while remaining nimble and responsive in a rapidly changing business environment.

So, clearly the choice of an operating model for a Data Mesh is a decision that shapes an organization's data future. It requires a careful balance between local autonomy and global coherence, between innovation and standardization. As organizations navigate this complex landscape, the key to success lies in aligning these choices with their long-term strategic vision, ensuring that their data ecosystem is not only responsive to current needs but also poised for future challenges and opportunities. So, clearly, the interplay between an organization's operating model and its data architecture is a fundamental aspect of effective data management and strategy. And, understanding this relationship helps organizations choose the right architecture to support their operational needs and strategic goals, whether it be a centralized data warehouse, a data lake, a data mesh, or a microservices architecture. For organizations looking to implement a data mesh, aligning its principles with their operating model, especially in federated structures, is key to leveraging its full potential.

Data Mesh as Loosely Coupled Regional Ecosystems

It is essential to approach the concept of Data Mesh as loosely coupled regional ecosystems with a moderately speculative

lens. While not a definitive outcome for all organizations, the regionalization of Data Mesh is a likely scenario that may unfold based on various factors such as organizational incentives, culture, and as discussed earlier, Conway's Law. This perspective should be seen as a "lesson learned"; unless proactive steps are taken to address it, regionalization could emerge as a natural trajectory for Data Mesh implementations.

The potential drift towards regionalized Data Mesh architectures (an example is shown in Figure 15-6, below) can be attributed to several driving forces. Organizational structures often shape the design and function of systems, including data architectures, as posited by Conway's Law. This law suggests that the communication patterns within an organization will likely reflect in its system designs. In the context of Data Mesh, this implies that organizations with distinct regional divisions might naturally develop data products that cater specifically to regional needs and regulations.

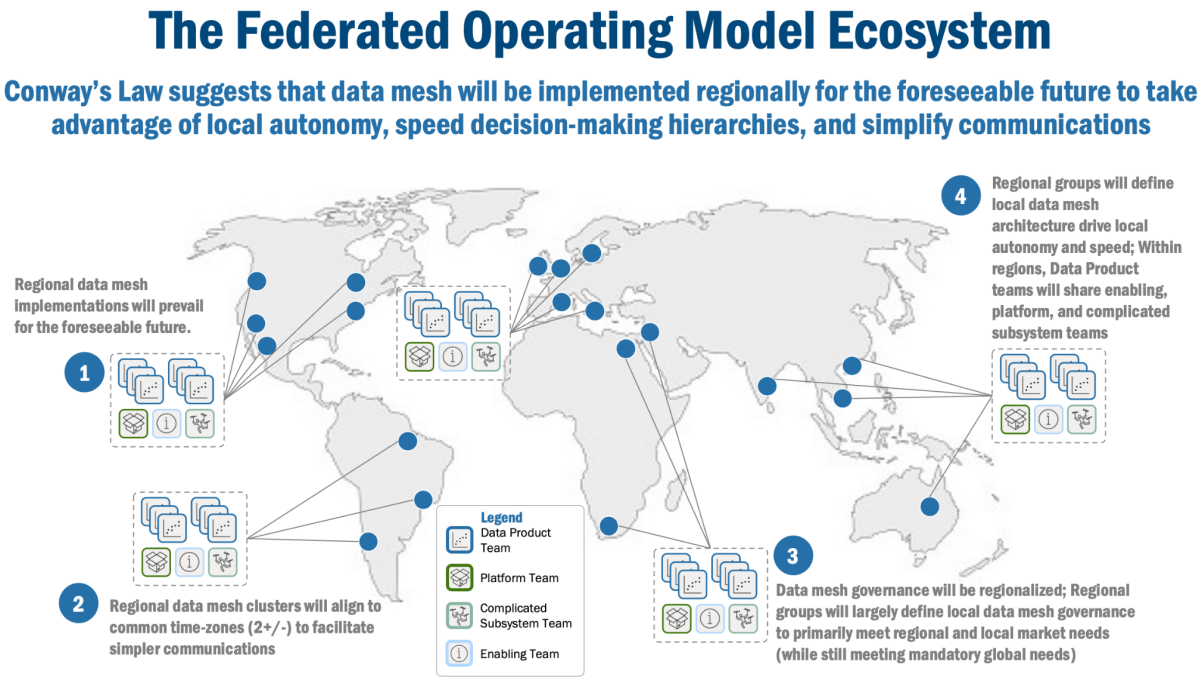


Figure 15-6. The Federated Operating Model Ecosystem

The regionalization of Data Mesh, while beneficial in addressing local market needs and regulatory demands, raises concerns about the cohesion of the organization's global data strategy. A regional focus may lead to highly tailored solutions that effectively serve specific areas but could also result in challenges when it comes to consolidating these varied data products into a unified global strategy. This could potentially lead to data silos and hinder efficient cross-regional data collaboration and sharing.

To counteract this natural inclination towards regionalization, organizations need to implement strategic measures. This involves fostering an organizational culture that values both

local autonomy and global integration. Mechanisms for inter-regional collaboration and knowledge sharing should be established, incentivizing teams to contribute beyond their regional scope.

Effective governance structures are crucial in this context. A hybrid governance model that combines overarching global policies with regional flexibility can ensure that local data products align with the broader organizational data strategy. Regular cross-regional meetings and forums can facilitate this alignment, helping to bridge the gap between local and global data objectives.

On the technological front, investment in infrastructure that promotes interoperability and seamless data exchange across regions is critical. This infrastructure must support diverse data types and formats, enabling effective communication and integration among various regional data landscapes.

So, clearly while the regionalization of Data Mesh is not a foregone conclusion, it is a possible outcome that organizations should be mindful of. By understanding the influences of organizational culture, incentives, and structural tendencies as suggested by Conway's Law, leaders can take preemptive steps to ensure that their Data Mesh strategy aligns with both

regional needs and global objectives. This balanced approach is key to realizing the full potential of Data Mesh as an effective tool for data management in a globally interconnected business environment.

All that being said, the evolution of Data Mesh into loosely coupled regional ecosystems may actually be a testament to the dynamic nature of data management in today's global business environment. This architectural shift, where data solutions are crafted to reflect the specific needs and contexts of various regions, is deeply rooted in the principles of Conway's Law. According to this law, the structure of systems developed within an organization is often a mirror of its communication patterns. In the realm of Data Mesh, this means that an organization's geographic or regional structure profoundly influences its data architecture, leading to the creation of region-specific data products that are both locally relevant and responsive.

The implementation of a Data Mesh as regional ecosystems requires a strategic approach that balances local needs with overarching organizational goals. This balancing act is crucial to ensure that regional data solutions, while tailored to specific market conditions and regulatory environments, also align with the global data strategy. It involves fostering a culture of collaboration and knowledge sharing among different regional

teams, enabling them to leverage unique regional insights while contributing to a unified data strategy.

Encouraging inter-regional collaboration is fundamental in this model. By designing incentive structures that reward cross-regional cooperation and the sharing of insights, organizations can cultivate a collaborative ethos. These incentives could range from recognition in organizational communication channels to financial bonuses, driving teams to not only excel in their regional domain but also contribute to the organization's collective data intelligence.

The governance of such a federated Data Mesh structure necessitates a delicate balance, and is discussed at length in the next chapter. However, suffice to say, each region must have the autonomy to address its unique challenges and opportunities, yet align with the central governance policies and standards that ensure consistency and integrity across the organization. This can be achieved through a hybrid governance model that combines universal core principles with region-specific guidelines. Regular forums or committees that bring together representatives from various regions can facilitate this, providing platforms for aligning strategies, sharing best practices, and addressing collective challenges.

Technology is the backbone of these loosely coupled regional ecosystems. A robust technological infrastructure, encompassing advanced data integration tools, cloud platforms, and APIs, is essential for seamless inter-regional data flow and collaboration. This infrastructure must be versatile enough to handle diverse data types and formats, ensuring effective communication and integration among regions with varying data landscapes. Such a technology stack not only supports the flexibility and scalability required for regional autonomy but also underpins the security and integrity vital for a cohesive global data strategy.

The transition towards regional Data Mesh implementations is a strategic response to the inherent diversity within large organizations. It aligns with Conway's Law by reflecting the decentralized nature of modern enterprises. However, the success of these regional ecosystems hinges on their ability to operate within a well-defined global framework. This necessitates a strategic vision that seamlessly integrates local autonomy with global data objectives, ensuring that regional data products contribute meaningfully to the organization's overall data strategy and business goals.

So, clearly the move towards regional Data Mesh models represents a sophisticated approach to data management, one

that respects the unique characteristics of different regions while maintaining a unified vision. By carefully balancing regional autonomy with centralized oversight and integrating technological solutions that facilitate collaboration and standardization, organizations can create a data ecosystem that is not only regionally effective but also globally coherent. This strategy ensures that organizations are well-positioned to harness the full potential of their data assets in a rapidly evolving global business landscape.

Chapter 16. Establishing a practical Data Mesh roadmap