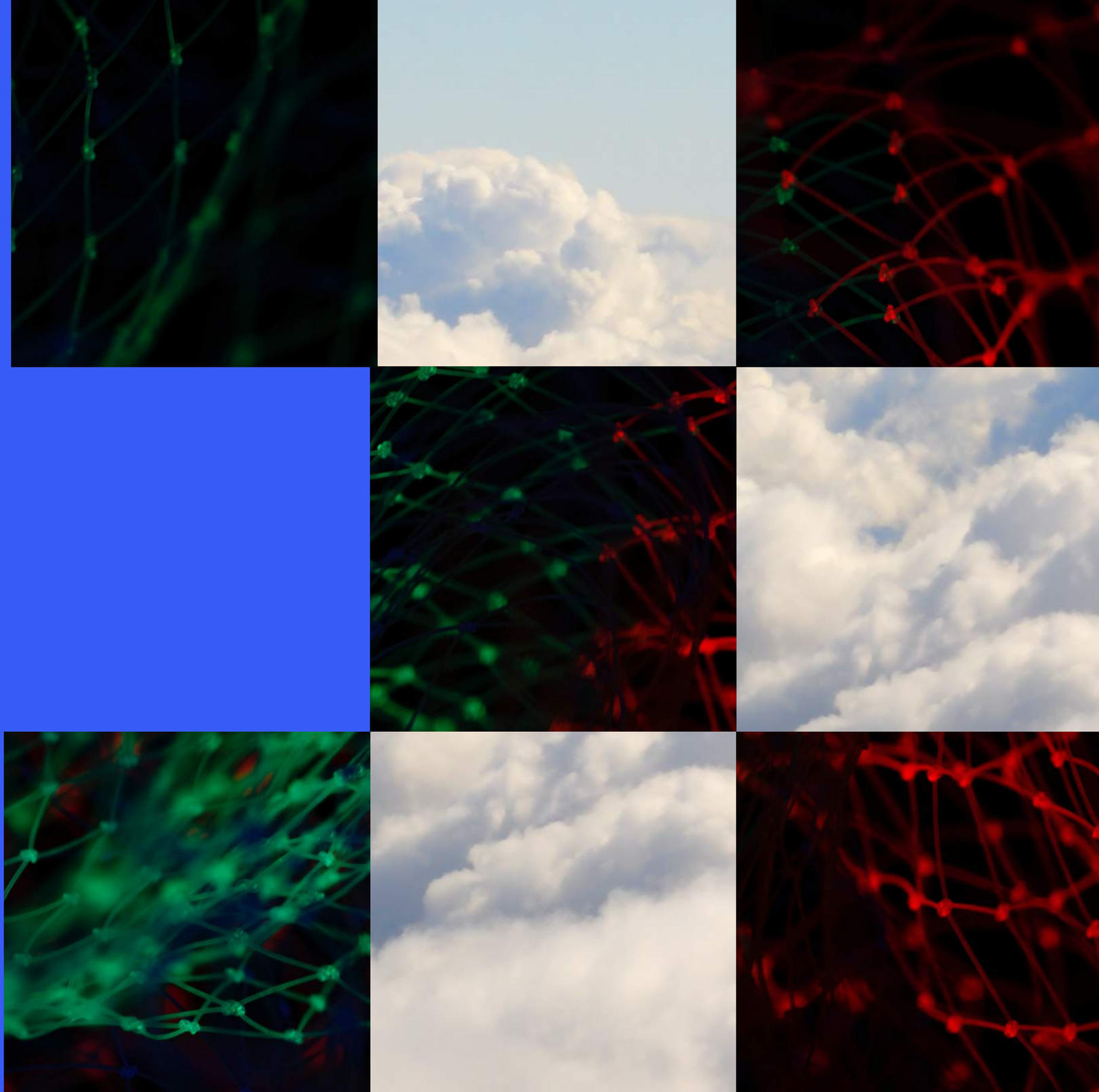
Datatonic Google Cloud

Data Mesh on Google Cloud

A guide by data leaders, for data leaders, looking to turn Data Mesh principles into practice on Google Cloud.



Datatonic

Introduction

The need for robust data and analytics has been an age-old focus for many organisations. Technology has evolved to meet the rapid growth in data volume, velocity and variety. Technology has also shifted to meet growing demands by organisations to utilise data for decision making.

We've witnessed these shifts over the last few decades, from:



1. Point solutions, where reports were generated out of operational databases, to



2. Vendor-centric data warehousing, where central data teams managed the ingestion, processing, and serving of data across the organisation, to



3. Centralised data lakes, where data was processed using "commodity" hardware (such as Hadoop and Map Reduce), to



4. Cloud-based solutions, which offer superior scalability, flexibility and reliability compared to previous on-prem solutions.

Technology has continuously adapted, however, the thinking behind having a single team responsible for creating and managing a monolithic data platform, has remained static - until now.

Datatonic

What is Data Mesh?

Data Mesh is a decentralised approach to data management. It champions data as a product, and empowers multiple business units to harness and share data more effectively, enabling businesses to address issues around speed, scale and ownership.

This paradigm follows a trend, observed more broadly in IT, of decomposing monolithic applications into smaller, more manageable microservices.

In the case of Data Mesh, the microservices concept applies to data, and is achieved through the use of data domains. A large, centrally-owned and managed data platform becomes many, smaller domain-specific data domains, managed by domain experts.

This shift of ownership from a central data team to data domains is a critical success factor.

About this paper

This whitepaper outlines the core principles and benefits of Data Mesh, and details how to implement Data Mesh on Google Cloud. Our aim is to complement existing literature (including those published by Thoughtworks and Google Cloud), and offer tactical advice on how to evolve from Data Mesh principles, to Data Mesh in practice. Data Mesh is not just a technical challenge but also an organisational one; this whitepaper covers practical tips for both of these apsects.

Note: We have removed the assumption of ML capability to establish a baseline for Data Mesh on Google Cloud - this will be the subject of a follow-up paper.

This shift of ownership from a central data team to data domains is a critical success factor of Data Mesh.

Authors



Andy Harding

Lead Cloud Architect



Nevile Reid

Practice Strategy Lead - Data & Analytics

Datatonic

Contents

Chapter 1: Introducing Data Mesh 05

Chapter 2: From principles to practice 09

Chapter 3: Data Mesh and data-driven culture 19

Conclusion 25

Contact 26

Datatonic

Chapter 1

Introducing Data Mesh

The four principles of Data Mesh

The four guiding principles of Data Mesh

Data Mesh was first introduced by Zhamak Dehghani in 2019. Her latest publication, “Data Mesh: Delivering Data-Driven Value at Scale” provides a solid framework for Data Mesh.

Before detailing our implementation approach on Google Cloud, we’ll firstly recap her four guiding principles, and provide commentary around the core benefits of each principle.

These principles are:

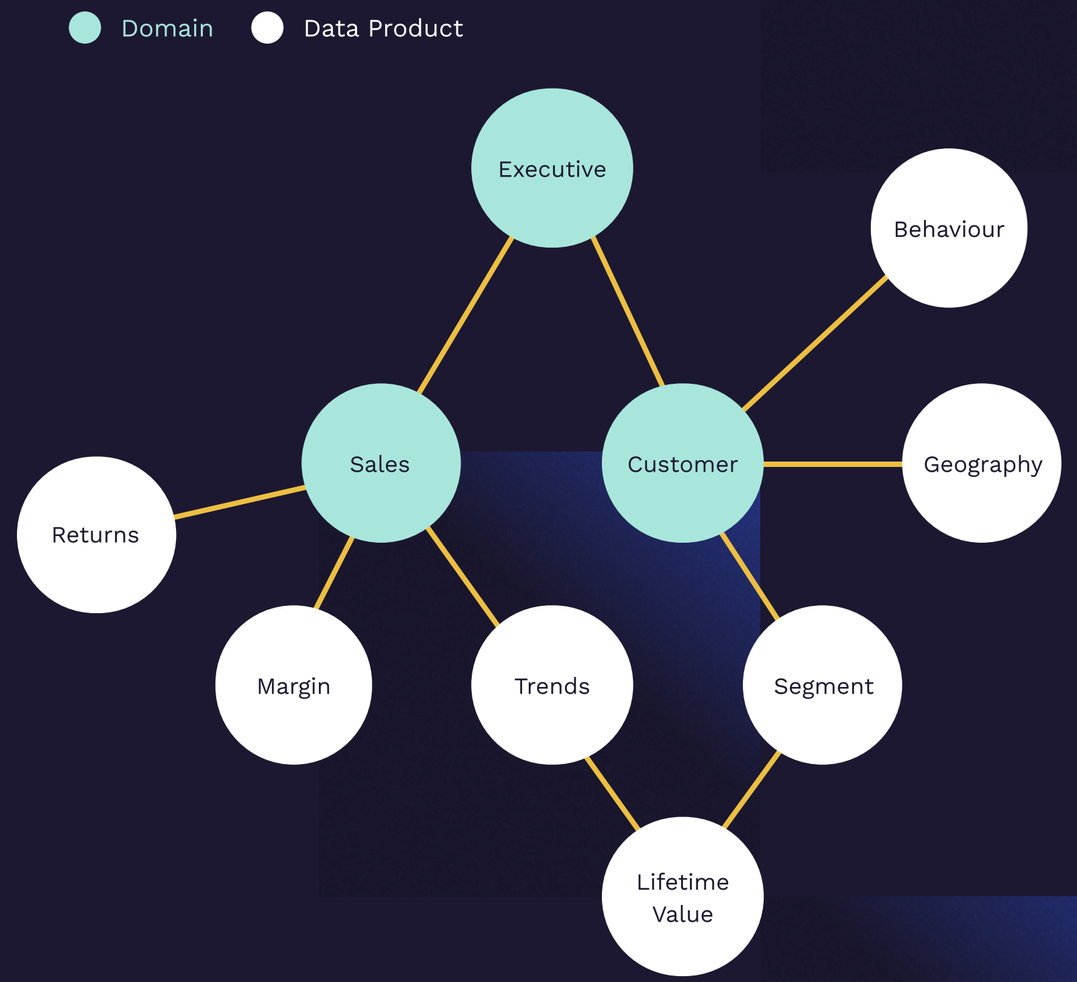
1. Domain ownership

2. Data as a product

3. Self-service data platform

4. Federated computational governance

Fig 1: Data Mesh - Data Domains vs. Data Products



Overall, these principles combine to support the creation of data products within multiple connected domains, and the subsequent sharing of these products between domains to create further data products for joint consumption. This process is illustrated in Fig 1 above.

Datatonic

Principle 1:
Domain Ownership

What it is

Like any successful initiative, ownership is a critical factor. Data Mesh promotes the use of individual, decentralised data domains. Data Products produced by domains therefore have clear ownership. This applies to both source-aligned data products and data products being created using data from other domains.

The core benefits

- Ownership is applied closer to where data is actually produced and consumed, enabling **rapid deployment of domain-focused analytics**.

- Domain-centric knowledge is more easily applied to the data and analytics development process, resulting in **valuable, trustworthy and impactful analytics** with closer collaboration between developer and consumer.

Principle 2:
Data as a Product

What it is

Data Products are the primary focus for data domain teams. Rather than simply focusing on creating data sets for single or tactical uses, Data Mesh proposes that “product thinking” is applied to encourage creation of purposeful and impactful data sets designed to drive insights. This is supported by consistent naming conventions, version change management, and user guides to ensure that data products are both discoverable and reusable across different domains.

The core benefits

- Data is more **discoverable, shareable, accessible**, and more **easily joined together,** thus increasing its use and value across the organisation.

- Treating data as a “product” drives a data culture **focused on value creation**, whilst ensuring data is **trustworthy and secure**.

Datatonic

Principle 3: Self-Service Data Platform

What it is

A self-service data platform provides an underlying technical platform of which all technical development related to the creation of data domains and products takes place.

This platform includes: storage, compute, data processing pipelines, discoverability, CI/CD, data quality, security, and privacy.

The core benefits

- Through robust templates, each domain team follows **best practices in platform engineering**, including **security and data governance**.

- These templates ensure **technical alignment and consistency**, and encourage **data and insight development to become central to each domain's business strategy**.

Principle 4: Federated Computational Governance

What it is

Data Mesh proposes the use of a federated governance structure which is made up of:

1. Representatives from each of the data domains e.g., Data Product Owners
2. Subject matter experts, such as those from compliance, security, legal, etc.

This cross-functional team is responsible for defining and agreeing policies, which are then handed off to the self-service data platform for computational enforcement across domains.

The core benefits

- Governance isn't managed by an independent team or across silos. Instead, **each domain has an input into the governance process and resulting policies**.

- Automation ensures that **policies are applied directly to the underlying technical platform for enforcement across all domains**.

Datatonic

Chapter 2

From principles to practice

Building Data Mesh on Google Cloud

Datatonic

There are key technical differences to focus on for a successful Data Mesh

Many of the core tools, techniques, and patterns used to build Data Lakes and Data Warehouses can be applied to building a Data Mesh. For example, the tasks related to ingestion from source systems, data transformation, lifecycle management, PII management, etc. remain largely the same.

We see the key areas of focus for a successful implementation of Data Mesh as follows:

1	2	3	4	5
Data Domains are provisioned by established DevOps and IaC approaches	Development templates accelerate the creation of data products	Self-Service Data Platforms must automatically enforce agreed governance policies	Domains must build Data Products and make them discoverable	Domains may consume Data Products from other domains

Next, we will drill into each of these areas and describe their implementation on Google Cloud using the following core technologies:



BigQuery
Analytical Data Warehouse



Data Catalog
Metadata Management



Analytics Hub
Data Exchange



Dataplex
Data Fabric

Datatonic

1

Data Domains are provisioned by established DevOps and IaC approaches



1. When creating a new domain, use standard DevOps tooling such as GitHub or GitLab **to launch predefined templates** supplied by the Self-Service Data Platform team. These are used to create Google Cloud projects, provision infrastructure and implement your policies.



2. To simplify this process, businesses can **deploy additional UI's over the DevOps pipelines** using tools such as Service Now, Env0 or Backstage. These tools provide a customisable UI to the Domain teams which can simplify their workflow and remove friction associated with learning how to use DevOps tools, such as GitHub and GitLab.



3. The DevOps pipelines **utilise IaC solutions**, such as Terraform, which are used extensively by Datatonic and clients. The Self-Service Data Platform team can create a standard set of Terraform modules that provisions infrastructure in a secure and compliant manner.



4. DevOps pipelines are then used, either directly or indirectly via a UI, by the Domain teams to **build out Data Products**. Pipelines also include linting, security scanning and testing features to ensure code remains consistent, high-quality and compliant, as well as implement policies which have been agreed upon by the Federated Governance model - more on this to follow.

Datatonic

2

Development templates accelerate the creation of data products

Development templates provide domain teams with a toolkit that allows them to accelerate the development of Data Products, and reside within the self-service data platform



1. Templates should be available for a **range of technologies,** since domains have autonomy over the tools they can choose, and cover a **breadth of functionality.** Functional coverage should include self-service data ingestion, data transformation, orchestration, training of ML pipelines, data modelling, data serving, etc.



2. Data domains should be given **autonomy** over which tools they can use **but not free reign**. It is a responsibility of the Federated Computational Governance team to define which tools sit within the Data Mesh toolkit. Only **approved tools** should be supported by the data platform and have templates created. This allows for some organisational strategy to be applied and to **reduce proliferation of tooling**.



3. The templates should also **cover common patterns** that are likely to be encountered when building data products. For example, in the case of data ingestion, they would cover technology specific ingestion, change data capture, backfill, delta and full loads.

Datatonic

3

Self-service data platforms must automatically enforce agreed governance policies

Defining governance policies is the responsibility of the Federated Computational Governance team. Once defined, the policies should be implemented within the Self-Service Data Platform for enforcement across all domains, and therefore all products within the Data Mesh.

Policies come in a few different types on Google Cloud:



1. Organisation Policies apply to Google Cloud projects according to an organisation hierarchy. These policies are useful for enacting broad governance controls such as:

- Limiting storage and compute to specific region(s)
- Enforcing security standards such as banning the use of external IP addresses
- Preventing storage buckets from being shared publicly

These types of policies should be implemented in Terraform like much of the code within the Self-Service Data Platform.



2. Data Access Policies determine which groups of users can access specific items of data. These can be set up using BigQuery and Data Catalog. For example, an organisation may define a data access hierarchy ranging from high sensitivity to low sensitivity.

The hierarchy levels can be used to determine which groups of users can see protected data assets (e.g., data in BigQuery) as clear, masked, or totally hidden from view. Much like the Organisation Policies, we recommend that data access policies are implemented and maintained in Terraform.

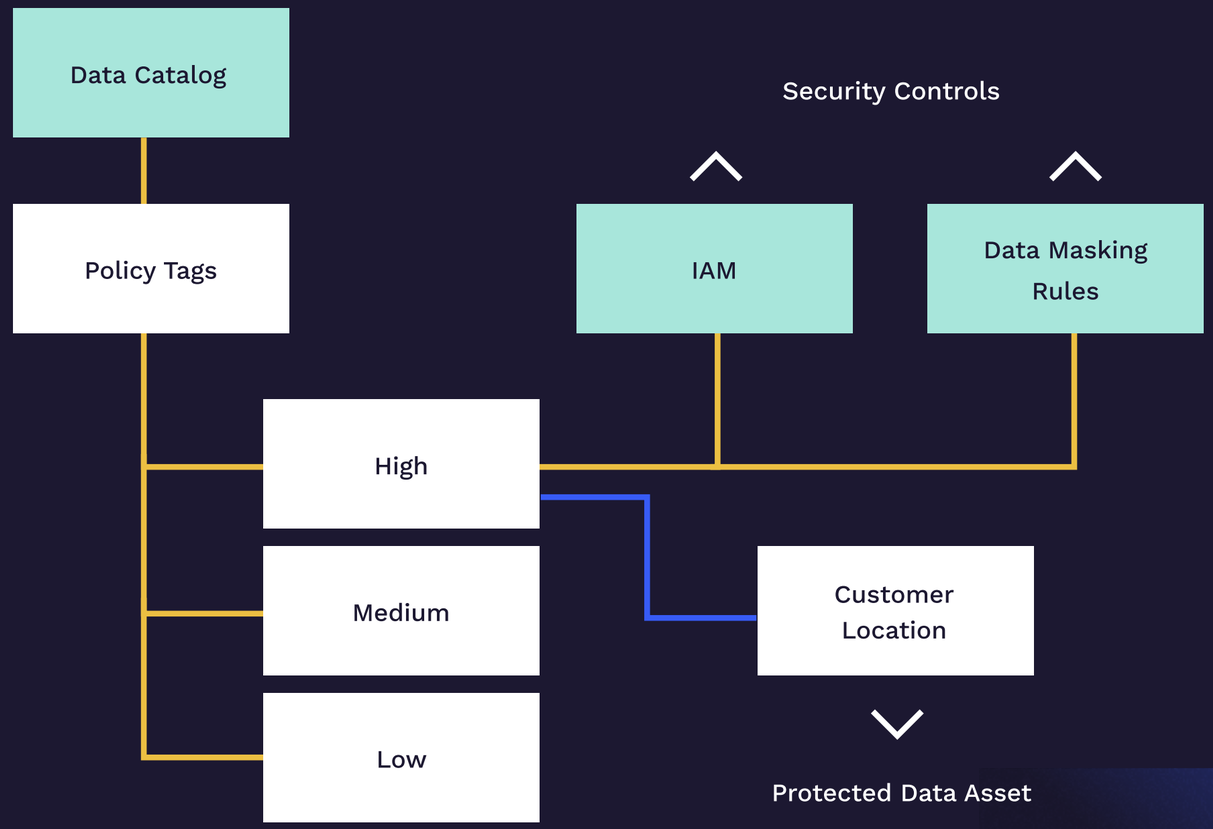
3

Self-service data platform must automatically enforce agreed governance policies

Policies and standards should also apply to data processing pipelines to ensure the practice of appropriately defining and applying robust data quality checks are in place and being actively used.

In addition to Google Cloud-specific controls, DevOps pipelines should also implement policy-based tools to ensure adherence to organisational standards and security best practices. Tools that could be used for this purpose include Checkov, Sonarqube and Open Policy Agent.

Fig 2: Hierarchy of policy and standards within data processing pipelines when using a Data Mesh.



*Both 1. And 2. are solved by Dataform and dbt on BigQuery

4

Domains must build data products and make them discoverable

As described earlier, Data Products have unique characteristics and successful implementation requires the following technical capabilities:

1. Data Products should be tested to ensure good data quality*

Datatonic often builds data pipelines that leverage the power of BigQuery i.e., by loading raw data and performing transformations within this technology. This adds the ability to **add data quality checks** e.g., uniqueness, null checks, format checks, as well as enhanced documentation and lineage.

3. Data Products should be shared and discoverable across the enterprise

Analytics Hub is a feature of BigQuery and allows an organisation to create a **dedicated data exchange**. Data Domains can publish Data Products into the data exchange where they become easily discoverable. The exchange can either be private, meaning data remains discoverable within organisational boundaries, or public, allowing an organisation to share data to external parties.

2. Business context (metadata) should be captured and applied to aid discoverability and understanding*

In recent years, we have seen the emergence of some key technologies which add huge value to this approach and are fully integrated with BigQuery. Examples include Dataform (in preview at the time of writing) and dbt. They provide **easy-to-define business metadata** such as table and column descriptions and the ability to define modules that can be reused across pipelines to increase discoverability.

4. It should be possible to attach good documentation so that Data Products are easy to use

This is also made possible using Google's Analytics Hub. When publishing a Data Product to the exchange, you can **attach metadata**, such as categorisation, publisher name, contact details, etc. In addition, you can attach free-text which can be used to fully document the Data Product or provide you a mechanism to link out to documentation hosted externally, such as on Notion, Confluence, etc.

Datatonic

4

Domains must build data products and make them discoverable

Documenting Data Products

When documenting Data Products, we recommend agreeing upon a structure for all domains to follow. This would be a deliverable that is agreed upon by the Federated Computational Governance team.

Influenced by Google Cloud's ML Model Cards, we propose a similar concept: "Data Product Cards". The central principle is that documentation should go beyond what is often done today e.g., focussing on transformation rules, granularity, refresh frequency, column description, and become more expressive and business-oriented.

Data Product Cards:

- Defining business impact
- Outlining the decisions that can be made using the data product
- Providing example code (e.g., SQL, Python) to perform analysis
- Explaining which visuals might be useful from an analysis perspective
- Adding performance tips
- Identifying limitations
- Communicating a clear roadmap
- Versioning and deprecation approach

Monitoring Domain Performance

When building Data Products, domain teams also need to focus on monitoring and publishing domain performance. This includes the reliability of data pipelines, data quality metrics and data access latency. Google Cloud provides a general solution to this problem through the use of Cloud Monitoring and its Site Reliability Engineering (SRE) principles which defines Service Level Indicators (SLI) and Service Level Objectives (SLO).

An example **SLI** would be pipeline reliability (good runs / total runs). This could *turn into* an **SLO**, such as "Pipeline reliability should be 99.8% over a rolling three-month period".

Once defined, Google Cloud will report on SLO performance and provide a mechanism for alerting should SLOs breach their targets. Like other features described here, SRE capabilities should be provided by the Self-Service Platform team for adoption by the Domain teams.

Datatonic

5

Domains may consume data from other domains

Analytics Hub makes the ability for Domains to consume Data Products from other Domains within the enterprise very straightforward. A member of the consuming Domain would browse the data exchange looking for the Data Product required. Once found, they add the Data Product directly from the exchange - or request access to it. Upon adding the Data Product, Analytics Hub automatically creates a new read-only linked data set (data is not physically copied) into BigQuery, taking care of permissions. Once the data set has been linked, the consuming Data Domain can query the data and is free to join, aggregate and otherwise integrate the data to create new higher order Data Products.

There are some other features of Analytics Hub worth mentioning:

1. Analytics Hub is currently used by Google to **host public datasets** such as Google Trends and Earth Engine which can be used to further enhance your Data Products.

2. Analytics Hub can be used to **share data externally** and according to the Google documentation provides monetisation opportunities in the future “As an Analytics Hub publisher, you can monetise data by sharing it with your partner network”. At the time of writing this, this feature was not yet available but it does provide an exciting opportunity for the future.

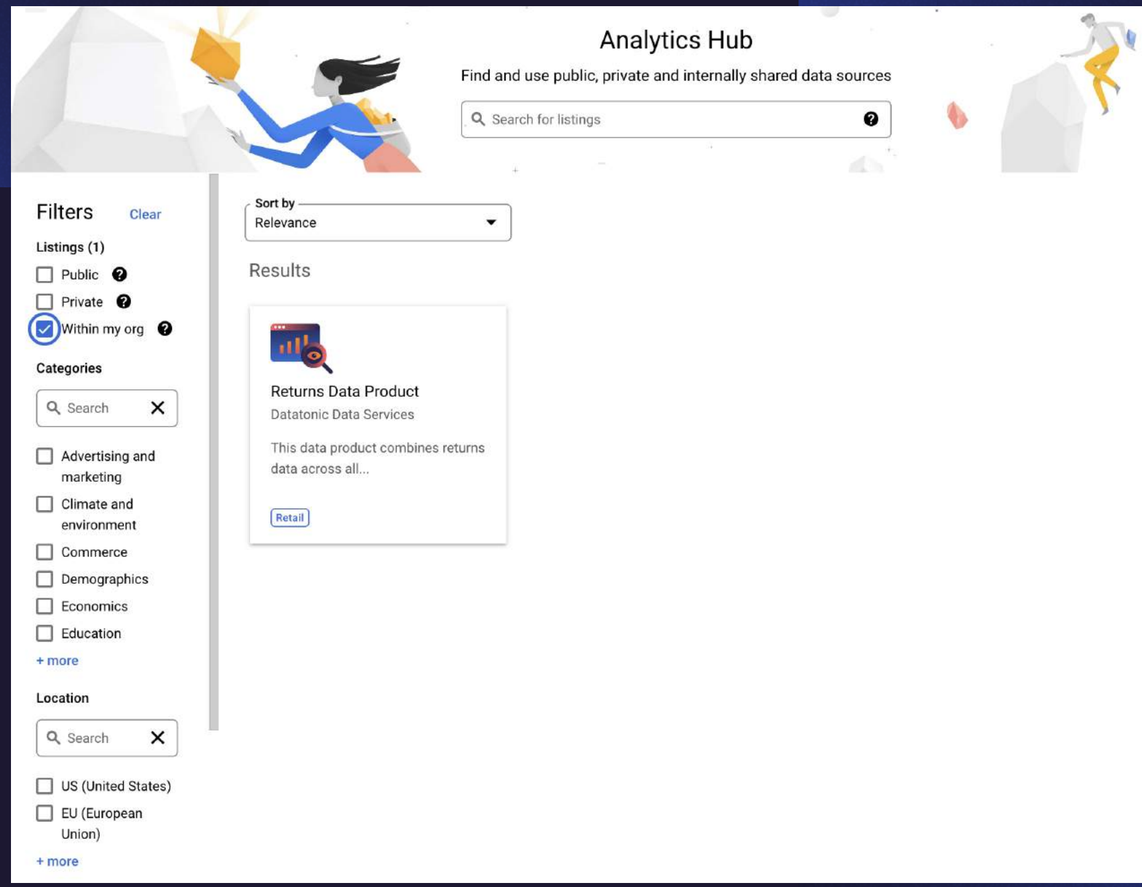


Fig 3: Analytics Hub listings can be used to discover data products published by other data domains within the organisation.

Reimagine existing data assets as a Data Mesh using Google Cloud's Dataplex

We've discussed BigQuery, Data Catalog and Analytics Hub as being key technologies for Data Mesh and have described how they can help solve the core challenges related to creating, sharing and consuming Data Products.

Google further compliments these technologies through its "**intelligent data fabric**" service called Dataplex. Dataplex acts as a **management layer** which **unifies distributed data, metadata, and access controls**. Data Catalog is a component of Dataplex.

Implementing Dataplex

The implementation of Dataplex involves the creation of logical data domains consisting of raw or curated zones. Zones can contain multiple data assets, such as objects on Cloud Storage or tables in BigQuery. Dataplex allows you to form these zones using existing assets providing a rapid route to deploying a Data Mesh.

Dataplex provides common integration templates covering activities such as file conversion, ingestion, and processing using tools such as Spark, Dataflow, etc. It supports robust data quality checking through the use of CloudDQ. Data that is managed by Dataplex is discoverable either through Data Catalog or through a Data Exploration Workbench allowing data to be explored and shared via tools such as Spark and Python within a Notebook-based environment.

Dataplex provides a compelling feature set as well as integration with other core Google technologies such as BigQuery, which are highly relevant to Data Mesh.

A raw zone supports in automating the load of raw data.

A curated zone prepares data to be analytics-ready by creating analytical files on Cloud Storage (e.g. Parquet, Avro) or tables in BigQuery.



Datatonic

Chapter 3

Data Mesh and data-driven culture

- Redefining Data Roles
- Tips for driving organisational change

Datatonic

Redefining Data Roles

New tooling and architectures will struggle to launch without a focused effort to manage the new processes and ways of working required by a Data Mesh.
This firstly involves understanding the **roles and staffing responsibilities** required to execute Data Mesh, some of which are an evolution of existing skills within a centralised data team.

Fig 4: Key Roles in Data Mesh

Role	Description	New or Refocused Role
Data product owner	• Accountable for Data Product(s) being produced by a Data Domain • Measures the performance of Data Products, and continues to make improvements	• **New role** - assuming the concept of Data as a Product is new to an organisation, Data Product Owners will be a new requirement. This role is key to ensuring the development of data products compliment the key principles and ways of working introduced by the Data Mesh
Domain data product developer(s)	• Accountable for the technical implementation of Data Products within a Domain • Utilises technology templates provided by the Self-Service Data Platform team	• **Refocused role** - Data Engineer or Analytics Engineer, or Data Scientists & ML Engineers for AI-based data product development
Platform product owner	• Accountable for the Self-Service Data Platform and ensures fit to organisational requirements and strategy • Ensures the platform caters to the needs of the Domain teams	• **Refocused role** - platform product owners should be focused and coordinated to ensure tooling is standardised across domains and aligned to the central strategy
Platform developer	• Accountable for the technical implementation of the Self-Service Data Platform	• **Refocused role** - Developers from existing data platform team including Data Engineers, Analytics Engineers, ML Engineers, DevOps Engineers, etc.
Data stewards	• Accountable for the processes within domains to enable the governance policies set by Federated Computational Governance. • Accountable for identification and implementation of data governance policies that are localised to the domain	• **Refocused role** - Data stewards continue to function but are aligned to Domains and are no longer centralised. Data governance policies are now a responsibility for Federated Computational Governance

+Datatonic

Consider staffing domain and platform teams with multi-disciplinary agile squads

In order to support the Data Mesh with the skills and needs required to develop a full range of data products, adopting agile squads and methodologies can be an effective response.

The following illustration shows how four different squads might be organised in order to build out data domains. You might start with the idea of developing a single squad per data domain, but for large and complex domains you could add more squads if needed. Like the Data Mesh, **the squad model is highly scalable** and can be replicated many times.

The exact makeup of the squad is determined by the **requirements and approach best suited to a specific domain**. Over time, the squad might evolve and develop new skill sets as underlying technology evolves.

This agile approach to data product development also encourages a multidisciplinary approach, leading to the **cross-pollination of skills and ways of working between engineering specialisms**.

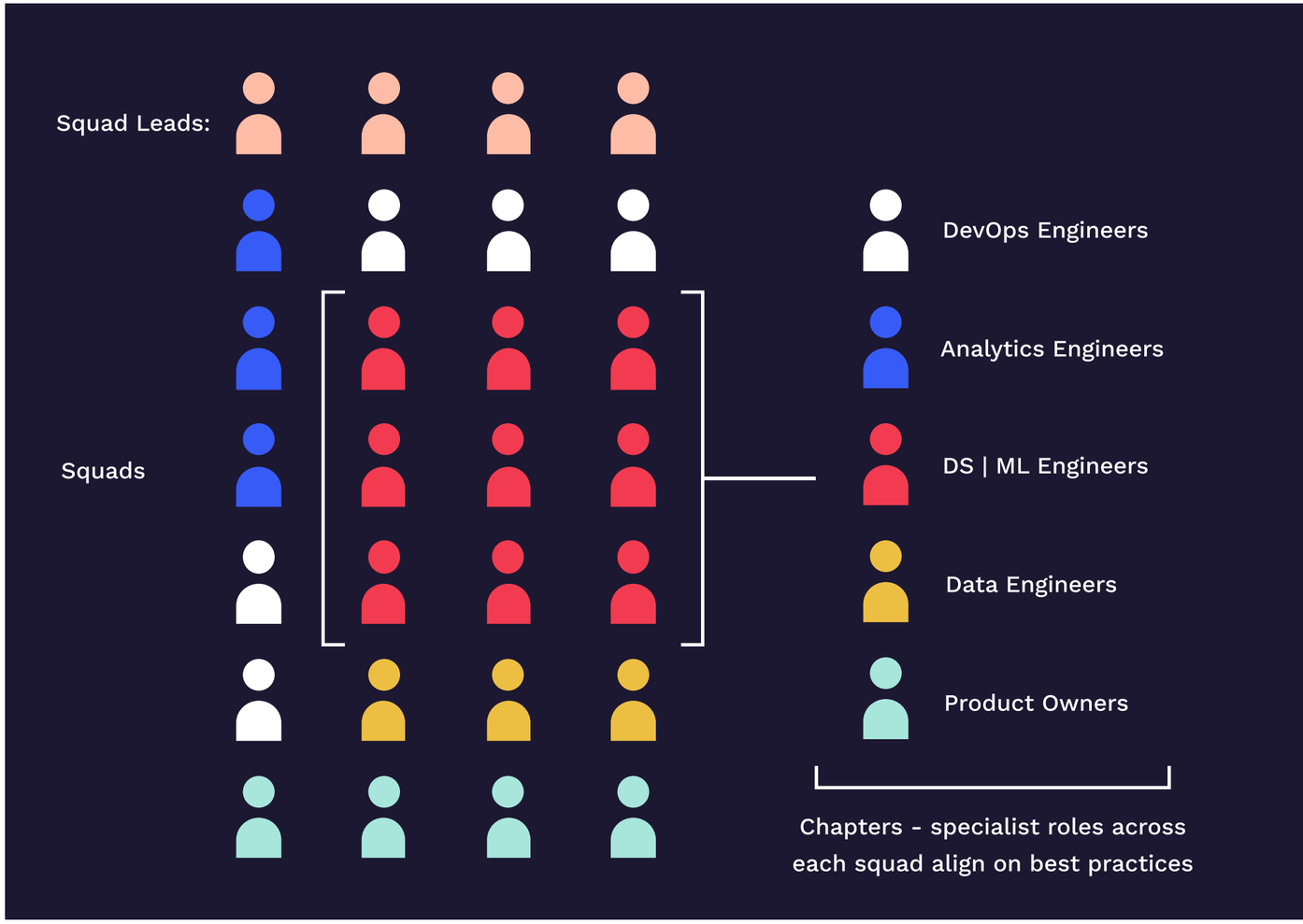


Fig 5: An Example of How Different Squads are Organised

Datatonic

Tips for driving organisational change

Despite the obvious benefits a move to Data Mesh may provide, it may not be immediately clear what steps are required to reach this future state. Below, we document **four key tips to manage the transition and adoption of a Data Mesh culture**.

Tip 1: Qualify first, Mesh isn't for all

Data Mesh targets specific pain points around **speed, scale and ownership**, which may not be the priority for all organisations at that given time.

Despite the growing interest in the paradigm, taking time to workshop and **qualify both the principles and feasibility of Data Mesh** will de-risk any resulting implementation. An initial evaluation will ensure key principles are considered, and any design or cultural change programme is rooted in this understanding.

An evaluation can also determine that a traditional, centralised model with optimisations, or a hybrid approach, may actually be a more effective route to value.

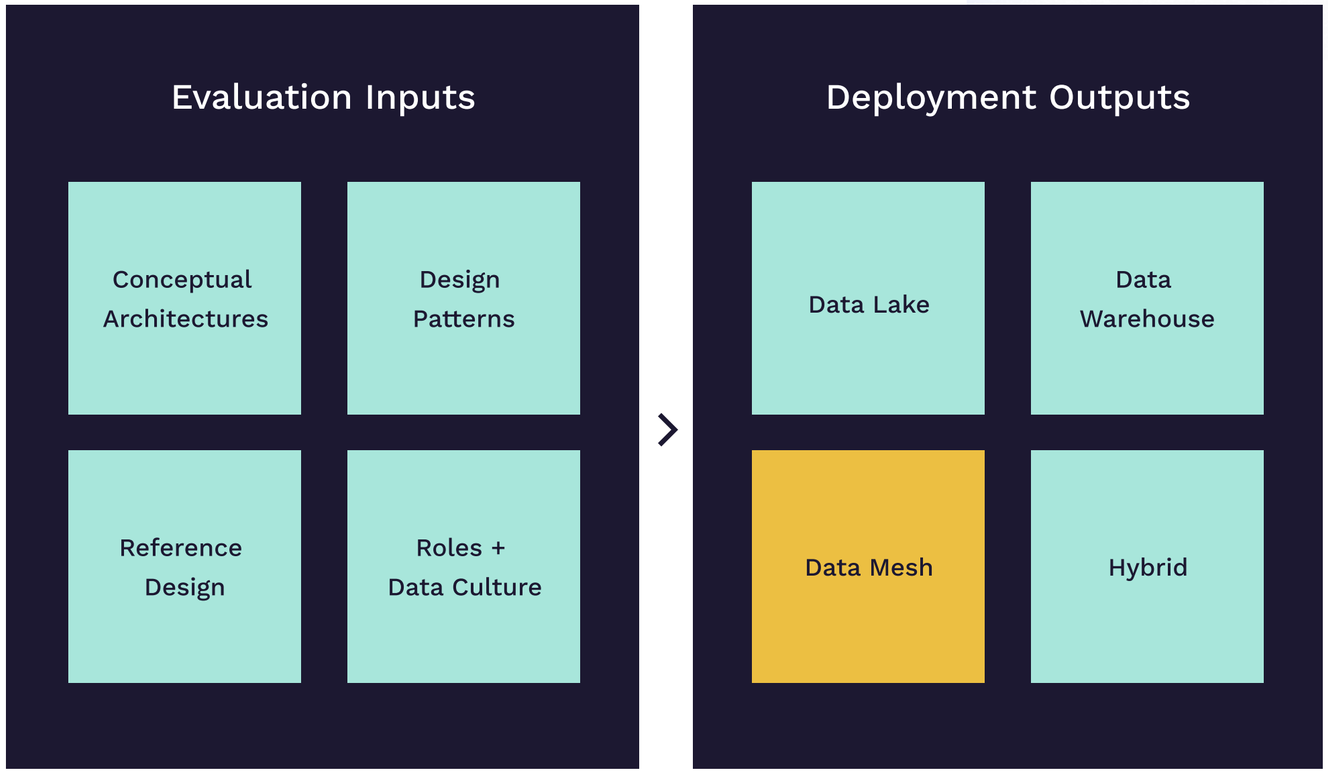


Fig 6: Inputs & Outputs of a Data Mesh Evaluation

Tip 2: Build an MVP to prove value fast

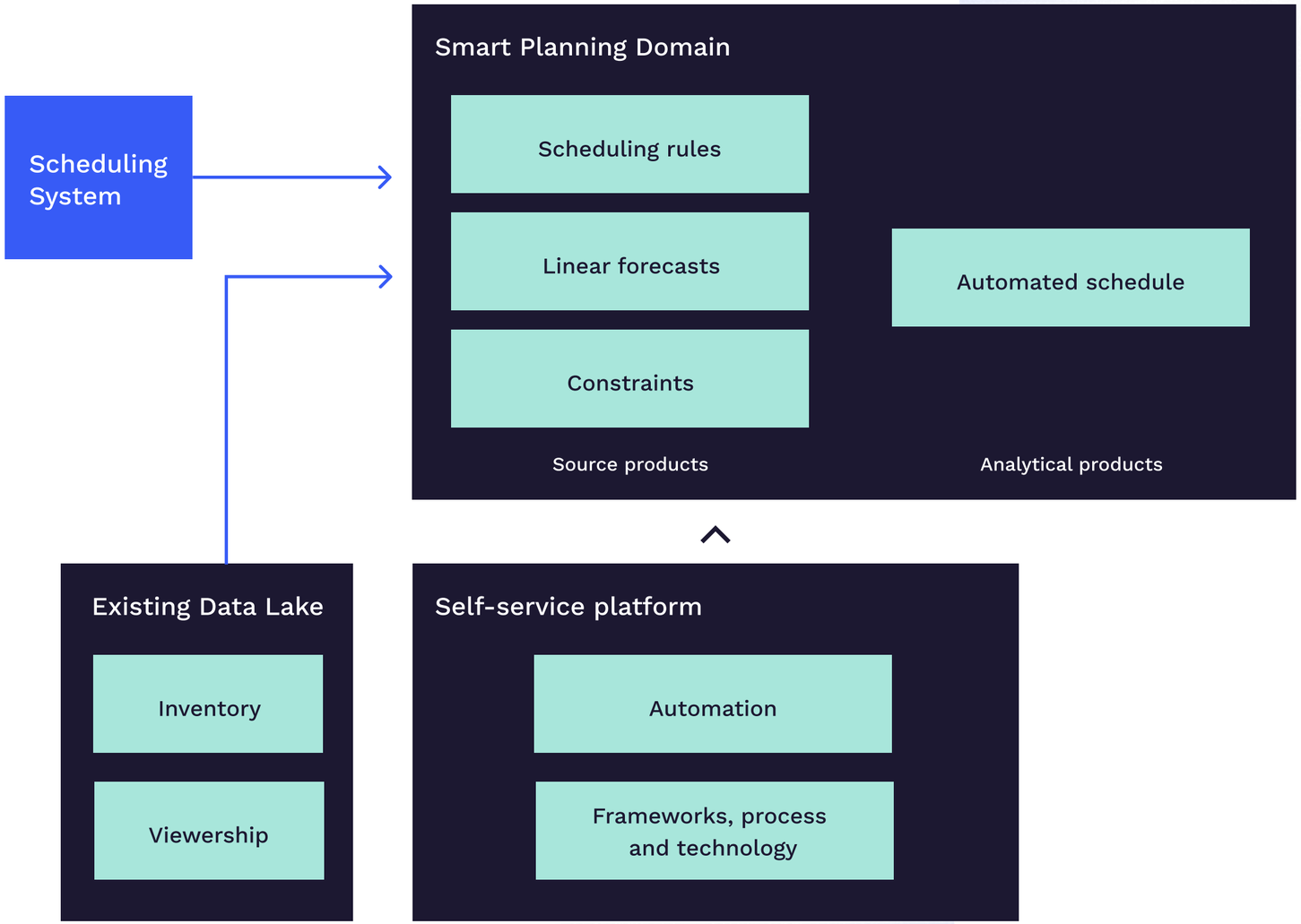
When it comes to building and implementing new data strategies and platforms, we strongly advise clients to take an MVP approach.

For Data Mesh, this means the creation of **one data domain**, including:

- Building the source data products and analytics data products for consumption
- Illustrating how users can discover, use and share data products
- Illustrating how data products will be governed

Through an MVP approach, organisations are able to **see and quantify the value of a wider Data Mesh adoption**, whilst testing new tooling and ways of working prior to rolling it out at scale.

Fig 7: MVP example for Media company - Smart Planning Domain



Datatonic

Tip 3: Incentivise owners + usage

To help promote new behaviours that champion the creation and use of good data products, some organisations have started to explore using financial, or other, means as an incentive.

This can also include incentivising the **use and creation of data products** to promote the following behaviours:

1. Focusing product developers on the products that will get the most use
2. Focusing product developers on the user experience and adoption of these products
3. Engendering proactive communication and distribution of the product
4. Creating a natural feedback loop for users, to strengthen and create better products

Tip 4: Consider secondments for the current central team

Organisations will grapple with how to redeploy previously centralised data development or engineering roles to a new decentralised or hybrid model.

During this transition, some roles may remain centralised to help manage the formation and management of the automated platform and tooling. Assuming organisations will activate new domains gradually, the scaling down and devolution of engineering and analyst roles will take time and require an element of double running to retain effective BAU.

One strategy to **assist a smooth transition** and **mitigate a need to significant upfront hiring** is the use of secondments, enabling the retention of a centralised team, whilst rotating some team members into new domains as they are set up and are brought into production.

This can be both effective in allowing existing team members to **test** their **interest in different domains**, whilst creating a **progressive knowledge transfer** for new ways of working & tooling from experienced central teams to the new domains.

Conclusion

Data Mesh presents an exciting opportunity to address some long-running pain points associated with the deployment of centralised data models. Data Mesh solves the challenge of scalability by decentralising the implementation of your data platform across a number of different Data Domains.

To fully adopt a successful Data Mesh, your organisation must embrace the four key principles: Domain Ownership, Data as a Product, Self-service Data Platform, and Federated Computational Governance. It's these principles, working together, that enable the benefits of a decentralised data model and guard against the risk of creating siloed domains.

Cloud technologies, such as Google Cloud, provide a compelling toolkit of services that can be used to implement your Data Mesh. Much of the technology foundations have been available for some time, but Data Mesh does require some specific tooling, particularly in the areas of creating and sharing Data Products and governance.

Implementation of Data Mesh should also be seen as a non-technical endeavour, as much as a technical one. Data Mesh requires increased coordination between data domains and requires broad enablement across your organisation. Data Mesh introduces new roles and responsibilities to your organisation and relies upon broad buy-in, as well as a mindset change for many.

Your organisation must embrace the four principles of Domain Ownership, Data as a Product, Self-service Data Platform, and Federated computational Governance. It's these principles, working together, that fulfil the promises being made by Data Mesh.

Datatonic

How to Get Started with Datatonic

Before embarking on Data Mesh, our recommendation is to spend time recognising the challenges that Data Mesh sets out to solve, and aligning these challenges with your own organisation.

Like any large scale transformational approach, Data Mesh may not be the right fit for everyone and can add complexity, so clarify the benefits and ROI upfront, before starting any transformation.

We further recommend conducting a MVP using Google Cloud to: (1) validate technology choices, and (2) make the Data Mesh real for your organisation and key stakeholders. Some of the technology on Google Cloud discussed here, e.g. Analytics Hub and Dataplex, are reasonably new and therefore require evaluation prior to implementation.

Datatonic is 4 X Google Cloud Partner of the Year, and can support you with a Data Mesh fit evaluation as well as building out an MVP ensuring full benefits can be achieved.

4 x Specialists on Google Cloud



hello@datatonic.com

London, United Kingdom
Gland, Switzerland
Stockholm, Sweden
Barcelona, Spain
Munich, Germany

Contact us →