DATA MESH ARCHITECTURE

MINI-PROJECT REPORT - TOPIC 8

DATA ENGINEERING SPECIALIZATION

RESIDENT: Tạ Ngọc Minh
BSc. student in Data Science & AI @ HUST

SUPERVISOR: Mr. Nguyễn Chí Thanh

Hanoi - 2023

PREFACE

Dear the board of directors of Viettel Group, the mentors, and all of my friends,

This document is a part of my research mini-project as a data engineer intern for the first period of Viettel Digital Talent 2023 Program. The completion of this research report does not mean the end of my duties as learners. In fact, this report is a first step towards consistently studying what I have reviewed and written here.

Data has grown up so quickly recently. The term 'big data' has become familiar with every tech-lovers is the clearest evidence of the explosion of data. This leads to many requirements of upgrading the current system and management, along with finding new architecture to store, process and analyze data. Once the central data lake has been overloaded, we need to design a decentralized architecture. This report is written under the research about the Data Mesh architecture, a new design that was first defined by Zhamak Dehghani in 2019.

To write this document, I would like to give many thanks to Viettel Group, and Viettel Digital Talent program for giving me chance to do my research. I want to send a big thanks to Mr. Nguyễn Chí Thanh, and all the mentors of the first period of the program, who gave me many useful lectures and guidance. Also, I want to say thanks to professor Huỳnh Thị Thanh Bình, professor Đỗ Tuấn Anh for their initial supports, and many other professors at the School of Information and Communication Technology, Hanoi University of Science and Technology about their public documents about data engineering. Last, I would like to say thanks to all of my friends, who always read my documents, comment about them and support me whenever I need.

Because this document is written in such a short time and some of the concepts are so new and hard to understand for a fresher like me, it is so hard to make sure it is completely correct, including the grammar errors and formatting errors in LaTeX. I would love to receive your contributions via **ngocminhta.nmt@gmail.com**.

Thank you for reading this report and looking forward to getting many comments from you.

Ngoc-Minh Ta

CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

Problems and Initial Approach

1.1 Real-life Problem

1.1.1 A problem from history

Many organizations have invested in building a central data lake. To make data-driven business, there is a central data administration team take the responsibility for this.

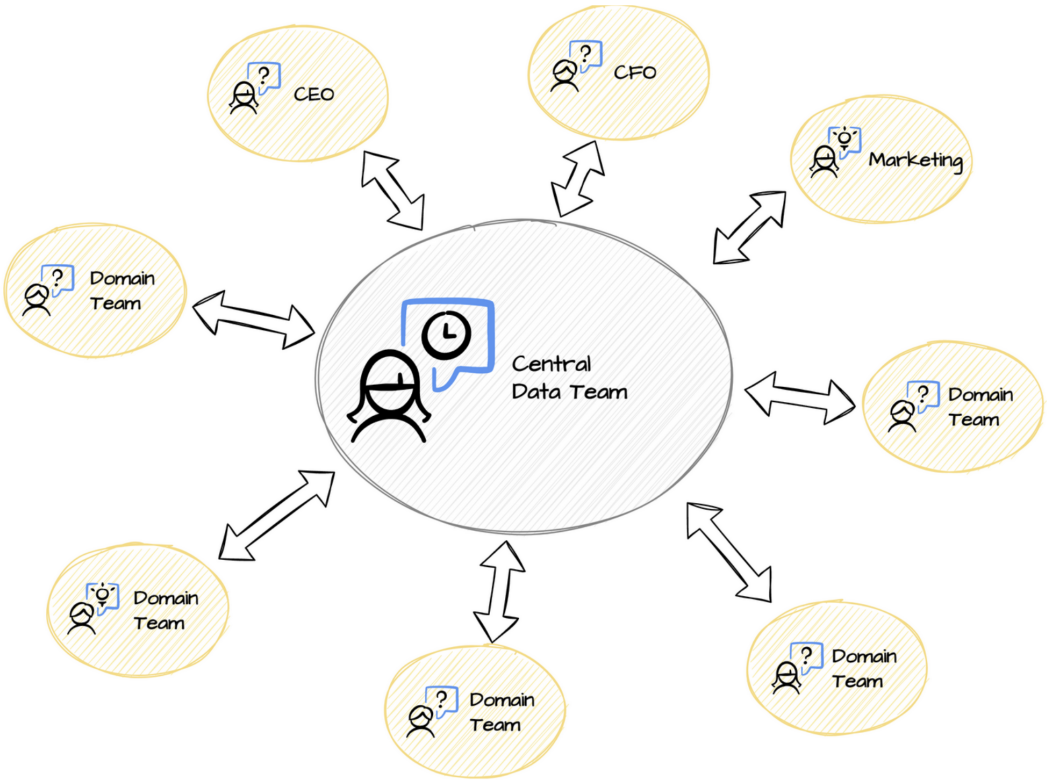

Figure 1.1: Central data team in a data-driven business

This model works quite well at the first time, however, after a while, they noticed that **the team often became bottleneck**. The team were overloaded with thousands of tasks and questions from multiple teams, along with the requests of time and accuracy. This lead to a massive problem since the competitiveness of business depends much on the speed of

analysis. For example, when we create a landing page for a new product, how does this change the influence clicking and buying rate?

But, why the response of the central data team is so slow and struggle? Actually, once the operational database changes, the team has to spend much of their time fixing the broken data pipelines. The little remaining time is insufficient for them to discover and understanding all the necessary domain data **for each question**. Getting the required domain expertise is a daunting task. [1, 2]

In contrast, some firms employ domain-driven design strategies that include independent domain teams and a decentralized micro-service architecture to relieve burden on the central data team. These teams exclusively control and understand their domain, including the company's information demands. The issue hasn't entirely been resolved, though. The domain teams must contact the overloaded central data team despite being aware of the key information needs and the domain, in order to obtain the essential data-driven insights. [1]

1.1.2 Demands from recent developments

Let's consider the scale-up of the software development over time. When one task grow bigger and bigger, we have to decentralize it into many smaller tasks, otherwise, all the organization will overload and lost of control. Take into consideration what we have done for the scale-up of software development:

- Decentralize business into domains;
- Decentralize engineering into autonomous teams;
- Decentralize monolith into micro-services;
- Decentralize operations into DevOps teams.

Hence, the next thing we need to do is scaling up data analytics by decentralizing data lake into data mesh.[2]

The position between domain teams and the central data team gets worse as the organization eventually grows. Transferring data management responsibilities from the central data team to the domain teams can help. Domain-oriented decentralization for analytical data is the central notion of the data mesh concept. Similar to APIs in a micro-service design, a data mesh architecture allows domain teams to conduct cross-domain data analysis on their own and connects data.

1.2 INITIAL APPROACH

1.2.1 What is Data Mesh?

The term data mesh was first stated by Zhamak Dehghani in 2019 and is based on four fundamental principles, which will be discussed in depth later:

- Principle of Domain Ownership;
- Principle of Data as a Product;
- Principle of the Self-Serve Data Platform;
- Principle of Federated Computational Governance.

1.2.2 What will changes after data mesh?

Data mesh is a fresh method for business intelligence, data analysis and management that is based on an innovative distributed architecture. Along with that, data lake and data warehouse do not disappear they just become nodes in the mesh. [3, 4]

Data mesh ensures organizations to continue to apply some data lake principles, such as making immutable data available for exploration or analytical use, and data lake tooling for internal implementation of data products or as part of the shared data infrastructure. Data lake would not be the centerpiece anymore.

It is an implementation detail sub-serving the idea of domain data product as the first-class concern. The same applies to data warehouse in terms of business reporting and visualization. [4]

Implementing a data mesh is not a purely technical project that we can implement in isolation from the rest of the business. It's not some thing that we can just start with, and then it works from the first try, but we have to grow and develop with it. It also cause a cultural shift in the organization.

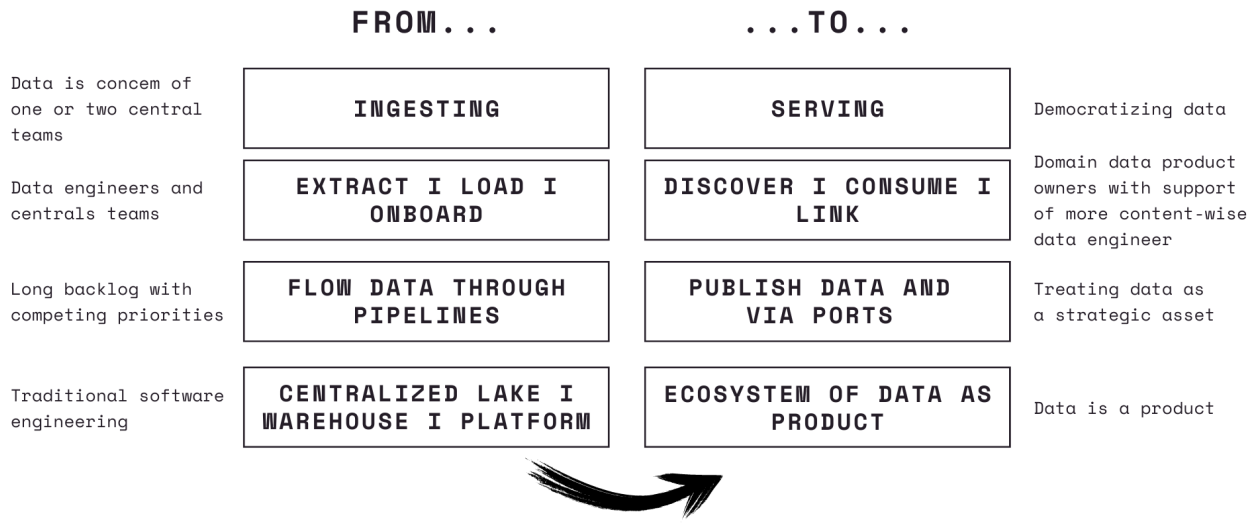

Figure 1.2: Cultural shift after implementing data mesh

Also, data mesh calls for a fundamental shift in the assumptions, architecture, technical solutions, and social structure of our organizations, that means, how we manage, use and own analytical data. [5]

Data mesh can be used as part of an enterprise data strategy, articulating the target state of the enterprise architecture as well as an organizational operating model with an iterative execution plan.

In its most basic form, it can be characterized by four interacting principles, which will be discussed in depth in section 2.1.

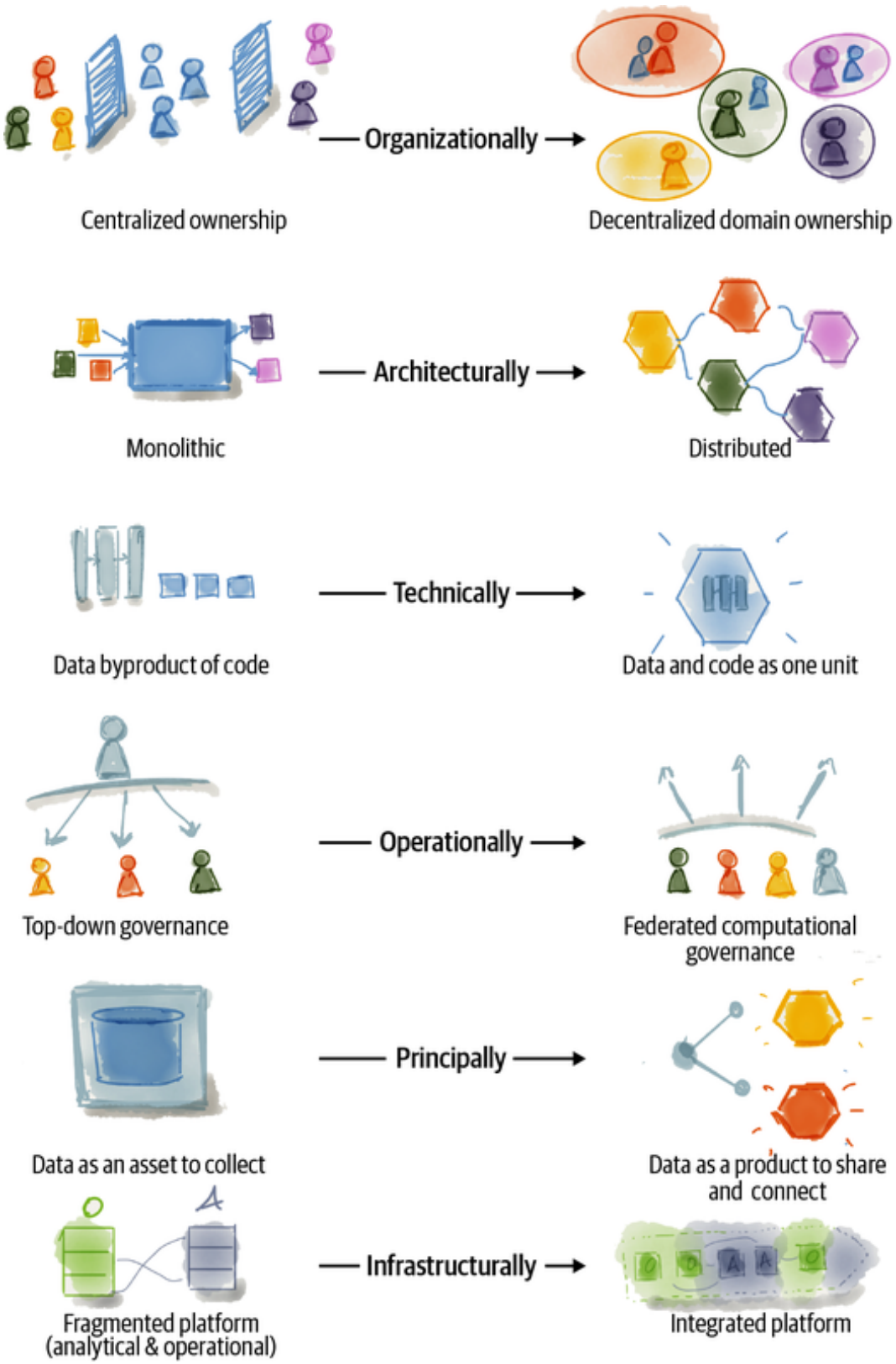

Figure 1.3: Data mesh dimensions of organizational changes

Chapter 2

DATA MESH ARCHITECTURE DESIGN

2.1 FOUR FUNDAMENTAL PRINCIPLES OF DATA MESH

Four simple principles can represent the logical architecture and operating model of data mesh. They are intended to move us closer to the goals of data mesh: increasing the value of data at scale, maintaining agility as an organization grows, and embracing change in a complex and turbulent business context.

2.1.1 Principle of Domain Ownership

Data mesh, at its core, is founded in decentralization and distribution of data responsibility to people who are closest to the data. This is to support a scale-out structure and continuous and rapid change cycles.

However, in contrast to traditional data structures with technological partition (e.g. data warehouse, data lake), data mesh follows *the seams of organizational units*. It follows the lines of division of responsibility aligned with the business using *domain-driven design (DDD) strategies*. Data mesh also gives the data sharing responsibility yo each of the business domains, each domain is responsible for the data it is most familiar with.

Domain-Driven Design (DDD) Strategies to Data

DDD is an approach to decomposition of software design and team allocation, based on the seams of a business. It defines a *domain* as "a sphere of knowledge, influence or activity."

In data platform architecture, the closest use of DDD is for source operational systems to emit their business domain events and the monolithic data platform to consume them. DDD's Strategic Design embraces modeling based on multiple models each contextualized to a particular domain, called a bounded context[1]. As Z. Dehghani's recommendation, data mesh adopts the boundary of bounded contexts to individual data products - data, its models, and its ownership. For some organizations have built services based on the domain bounded contexts, now, they apply the same decomposition and modeling to their analytical data in each domain.

[1] A bounded context is the delimited applicability of a particular model that gives team members a clear and shared understanding of what has to be consistent and what can develop independently. [6]

Domain data ownership is the foundation of scale in a complex system like enterprises today. When we map the data mesh to an organization and its domains, we discover a few different archetypes of domain-oriented analytical data, then we defined them as:

- Source-aligned domain data (native data product): Analytical data reflecting the business facts generated by the operational systems.
- Aggregate domain data: Analytical data that is an aggregate of multiple upstream domains.
- Consumer-aligned (fit-for-purpose) domain data: Analytical data transformed to fit the needs of one or multiple specific use cases.

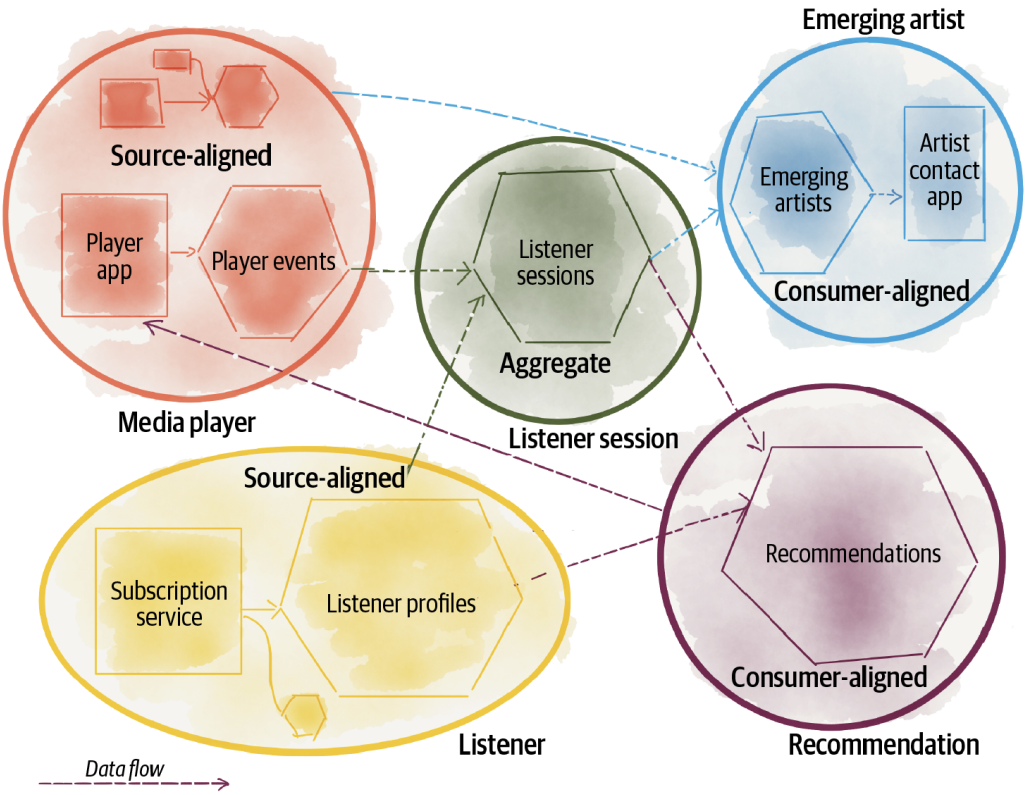

Figure 2.1: Decomposing the analytical data ownership and architecture, aligned with business domains, along with their data archetypes.

Chapter 3

DATA PRODUCT DESIGN & IMPLEMENTATION

Chapter 4

DATA MESH IN USE

4.1 DATA MESH WITH DATA LAKEHOUSE

4.2 FRAMEWORKS AND TECHNOLOGIES FOR DATA MESH

4.3 CASE STUDY OR DEMO

REFERENCES

[1] Z. Dehghani, "Prologue: Imagine data mesh," in *Data Mesh: Delivering Data-Driven Value at Scale*. O'Reilly Media, 2022, pp. xxv–xxxviii.

[2] C. Jochen, V. Larysa, and S. Harrer. (2023) Data mesh from an engineering perspective. [Online]. Available: https://www.datamesh-architecture.com/

[3] I. A. Machado, C. Costa, and M. Y. Santos, "Data mesh: Concepts and principles of a paradigm shift in data architectures," *Procedia Computer Science*, vol. 196, pp. 263–271, 2022.

[4] D. Nicolas. (2023) Will the buzz around data mesh really help solve my data challenges? [Online]. Available: https://kpmg.com/be/en/home/insights/2023/03/lh-the-impact-of-data-mesh-on-organizational-data.html

[5] Z. Dehghani, "Data mesh in a nutshell," in *Data Mesh: Delivering Data-Driven Value at Scale*. O'Reilly Media, 2022, pp. 3–14.

[6] E. Evans, *Domain-Driven Design: Tackling Complexity in the Heart of Software*. Addison-Wesley Professional, 2003.