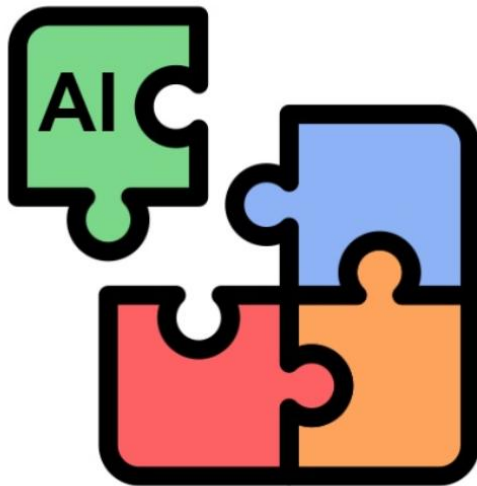


Data-Oriented Architectures

Christian Cabrera
University of Cambridge
08/11/2024

Previously...

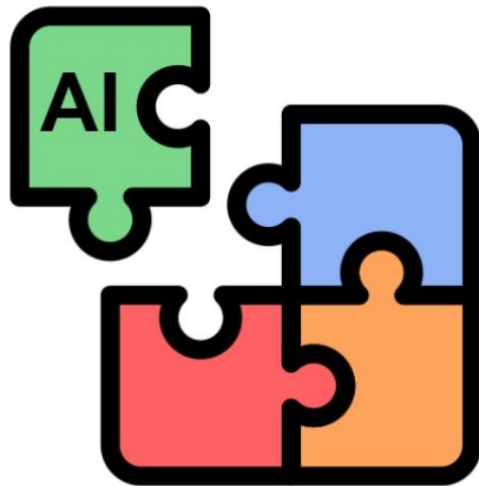
- The Systems Engineering Approach



**Software
Systems**

Previously...

- The Systems Engineering Approach

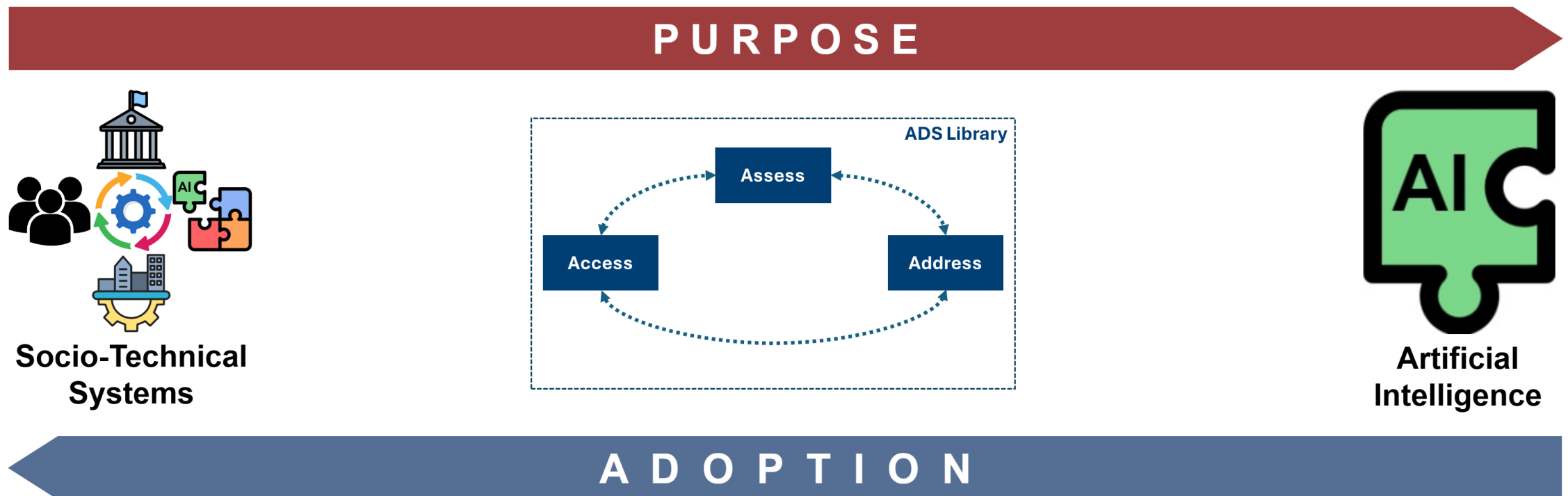


Software
Systems

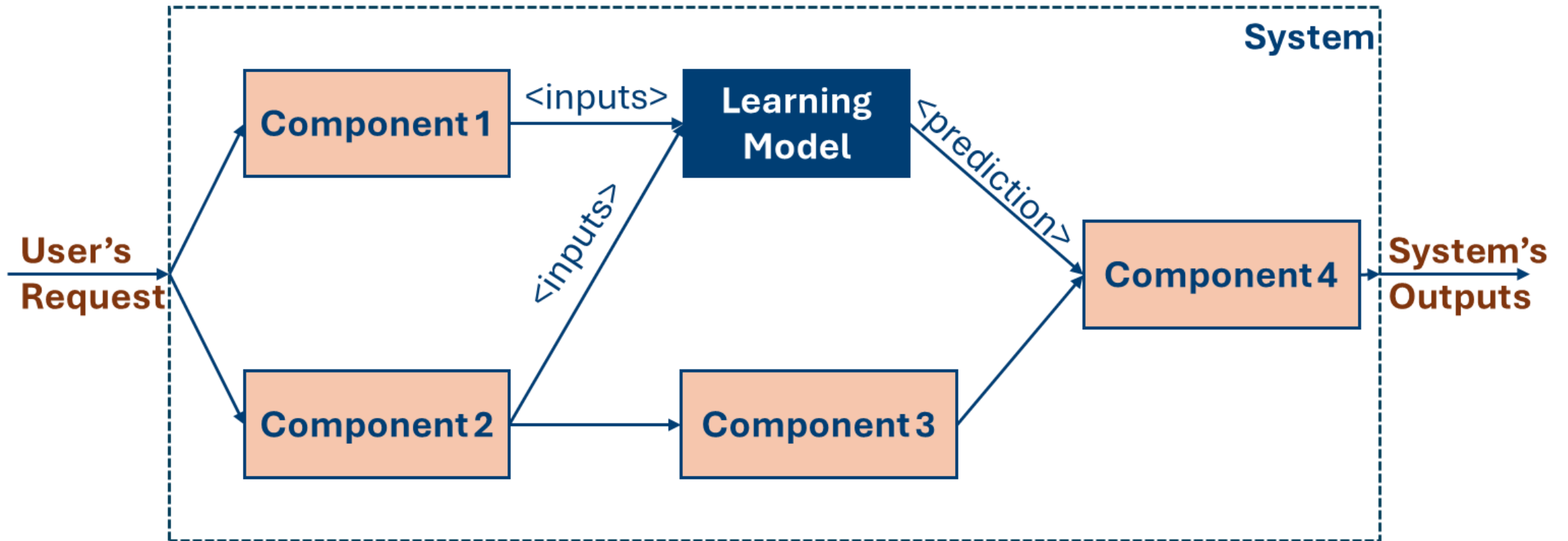
- **Systems Thinking:** System Views, Agility System, and System Dynamics.
- **Process Model:** Top-Down Analysis, Variant Creation and Problem-Solving Cycle.

Previously

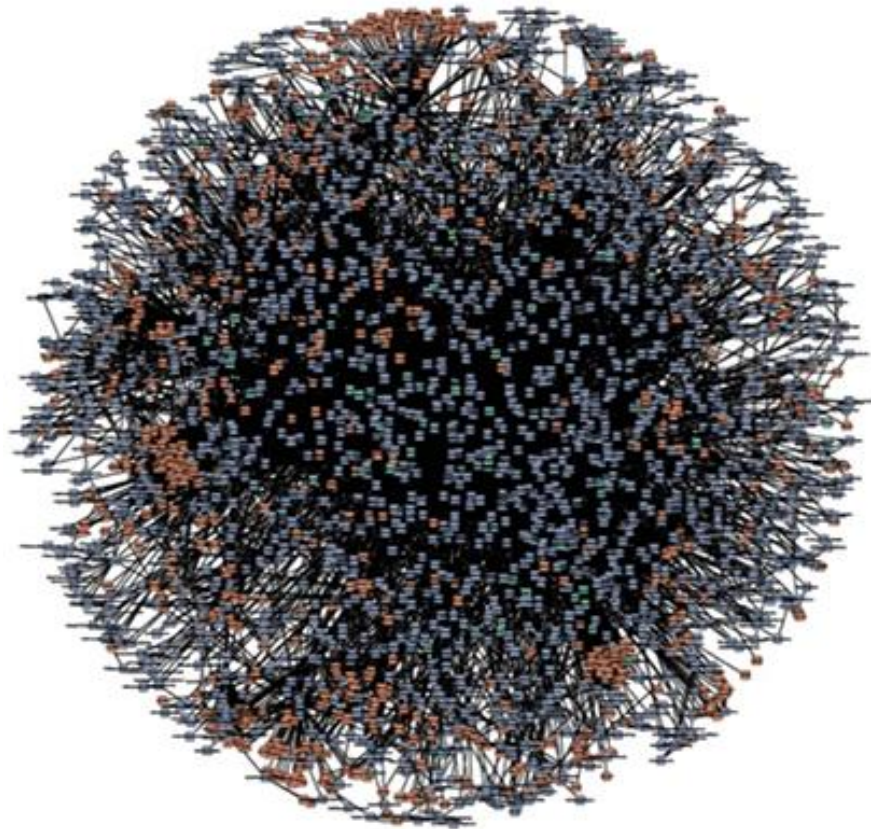
- The Systems Engineering Approach



Real-world Deployments



Real-world Deployments



amazon.com[®]



Source: <https://www.divante.com/blog/10-companies-that-implemented-the-microservice-architecture-and-paved-the-way-for-others>

Systems Design

Systems' design decisions change from system to system...

Systems Design

Systems' design decisions change from system to system...

**But we can identify common requirements between
software systems.**

Systems Design

Systems' design decisions change from system to system...

**But we can identify common requirements between
software systems.**

**Based on these commonalities we can define design
patterns and systems architectures.**

Systems Design

Systems in the age of the Internet required:

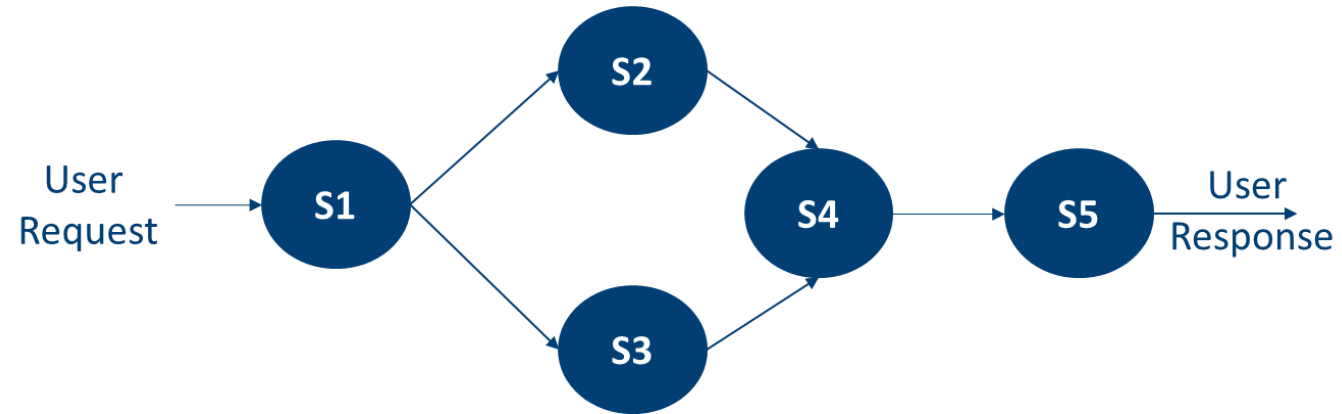
- Separation of concerns
- High availability
- Scalability
- Low Latency

Systems Design

Systems in the age of the Internet required:

- Separation of concerns
- High availability
- Scalability
- Low Latency

Service-oriented Architectures (SOAs)

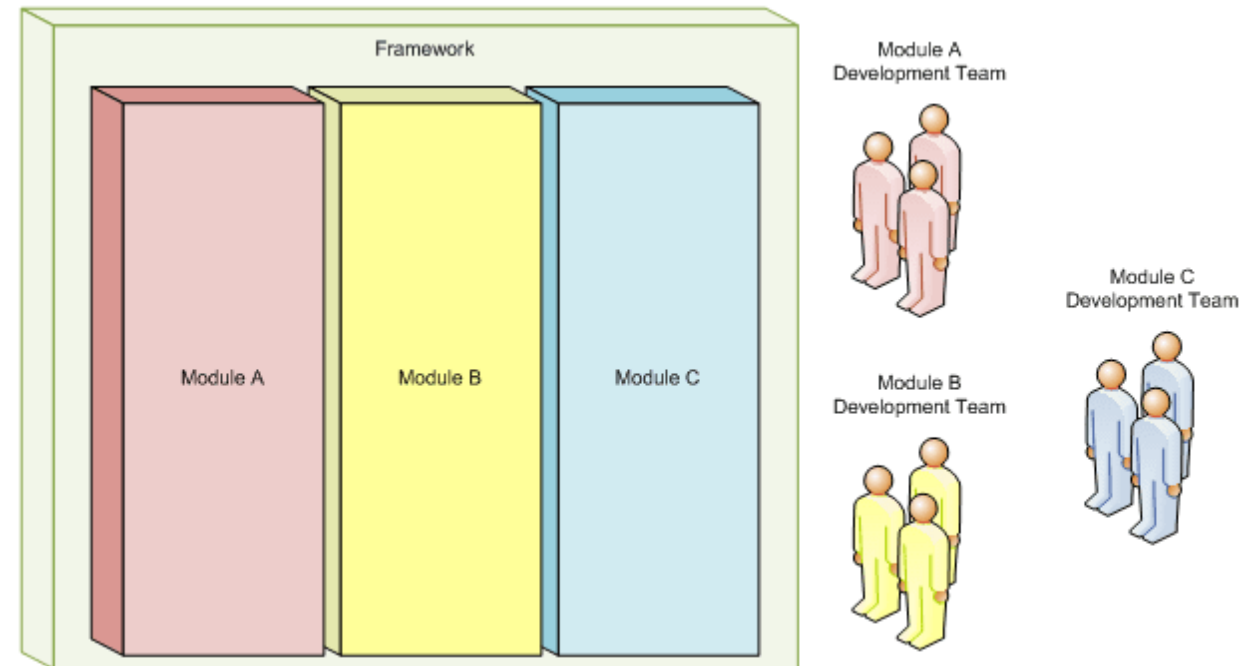


Systems Design

Systems in the age of the Internet required:

- **Separation of concerns**
- High availability
- Scalability
- Low Latency

Service-oriented Architectures (SOAs)



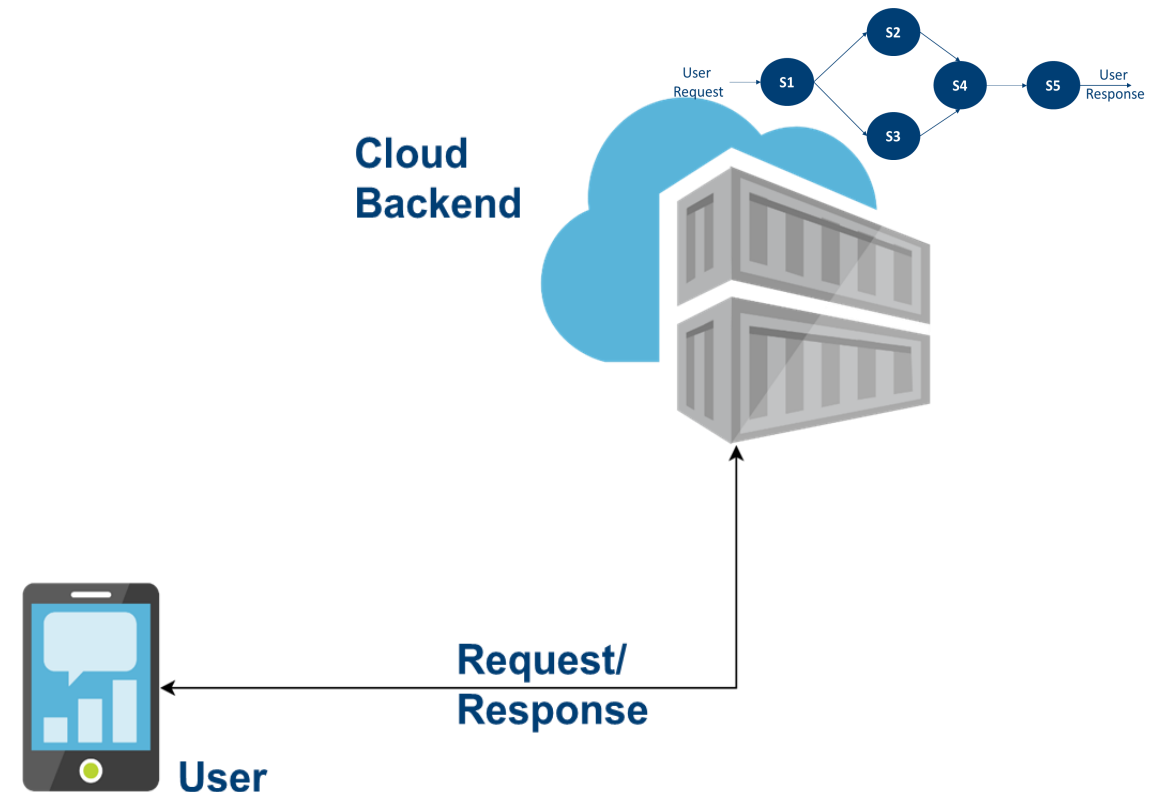
Source: <https://dev.to/suspir0n/soc-separation-of-concerns-5ak7>

Systems Design

Systems in the age of the Internet required:

- Separation of concerns
- **High availability**
- **Scalability**
- **Low Latency**

Service-oriented Architectures (SOAs)



AI Systems Design

Systems in the age of AI are data-driven:

- Data availability
- Data ownership
- Data traceability and monitoring
- Super-low latency requirements
- Sustainability

AI Systems Design

Systems in the age of AI are data-driven:

- Data availability
- Data ownership
- Data traceability and monitoring
- Super-low latency requirements
- Sustainability

Service-oriented Architectures (SOAs)

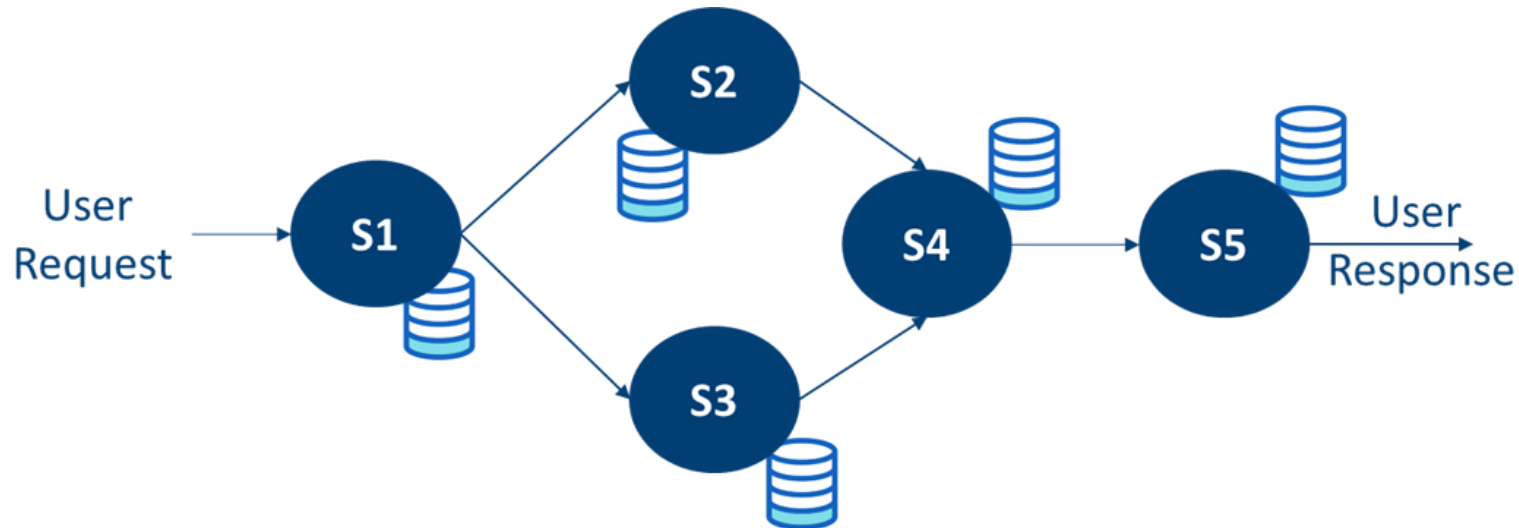
The Data Dichotomy:

“While data-driven systems are about exposing data, service-oriented architectures are about hiding data.” [1]

AI Systems Design

The Data Dichotomy:

“While data-driven systems are about exposing data, service-oriented architectures are about hiding data.” [1]



AI Systems Design

The Data Dichotomy:

“While data-driven systems are about exposing data, service-oriented architectures are about hiding data.” [1]

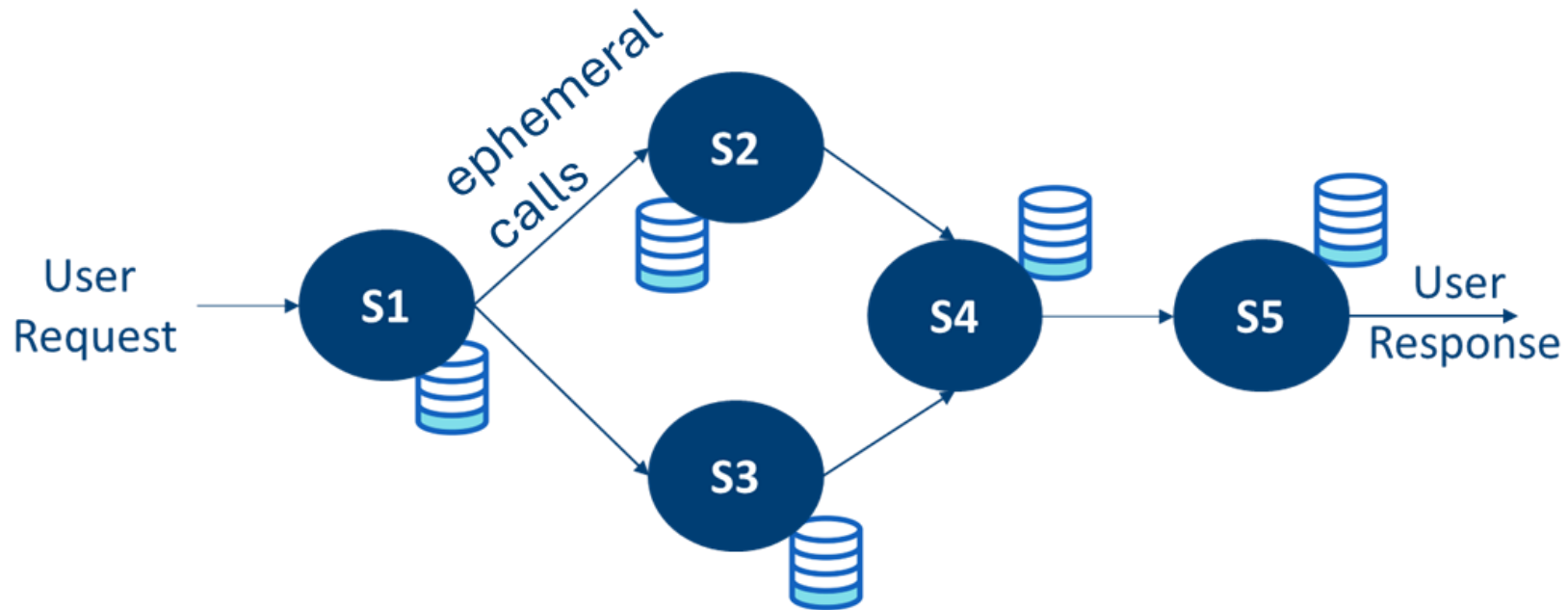
We need to design systems prioritising data!

Data-Oriented Architectures

- Data-first systems
- Prioritise decentralisation
- Openness

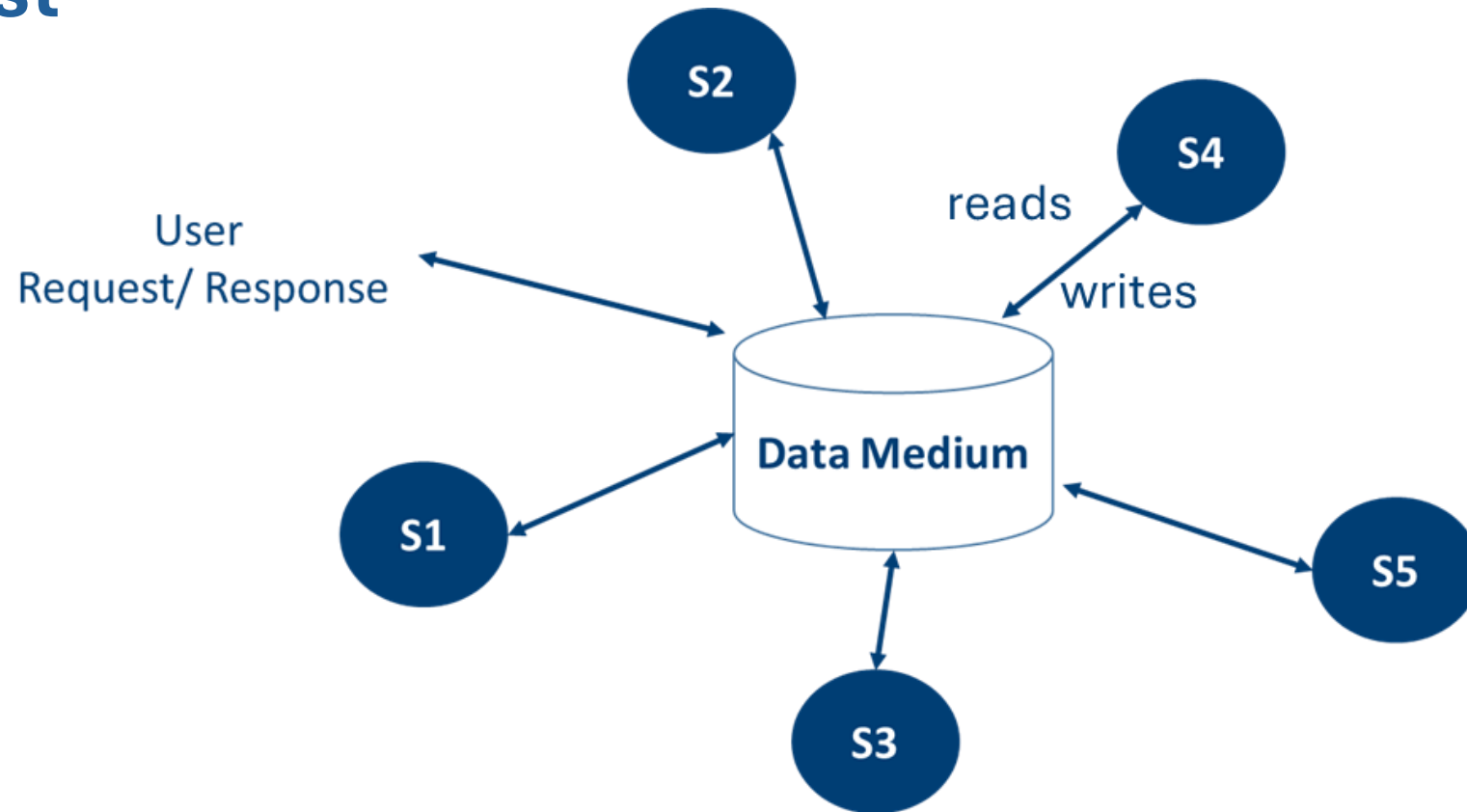
Data-Oriented Architectures

Data-First



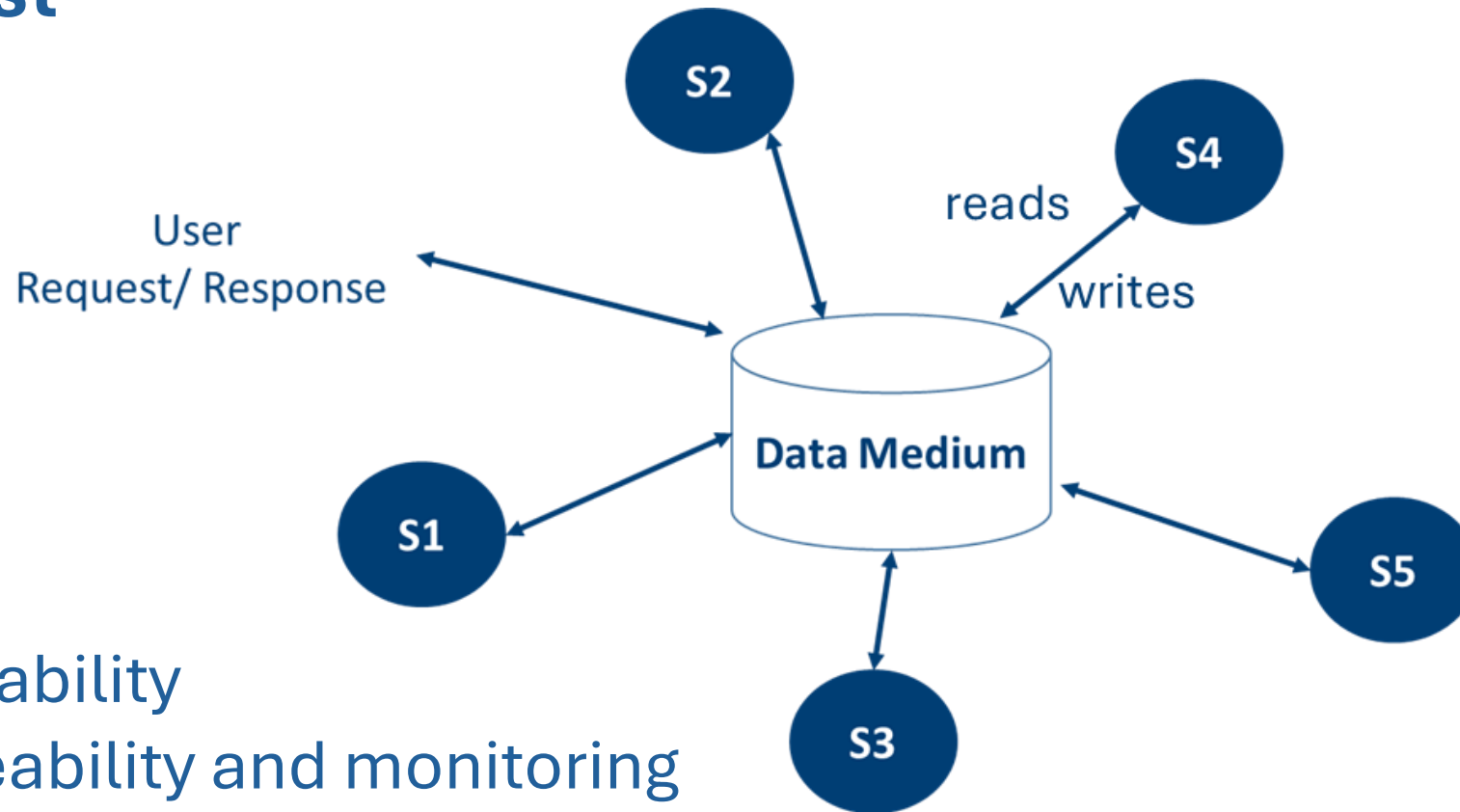
Data-Oriented Architectures

Data-First



Data-Oriented Architectures

Data-First

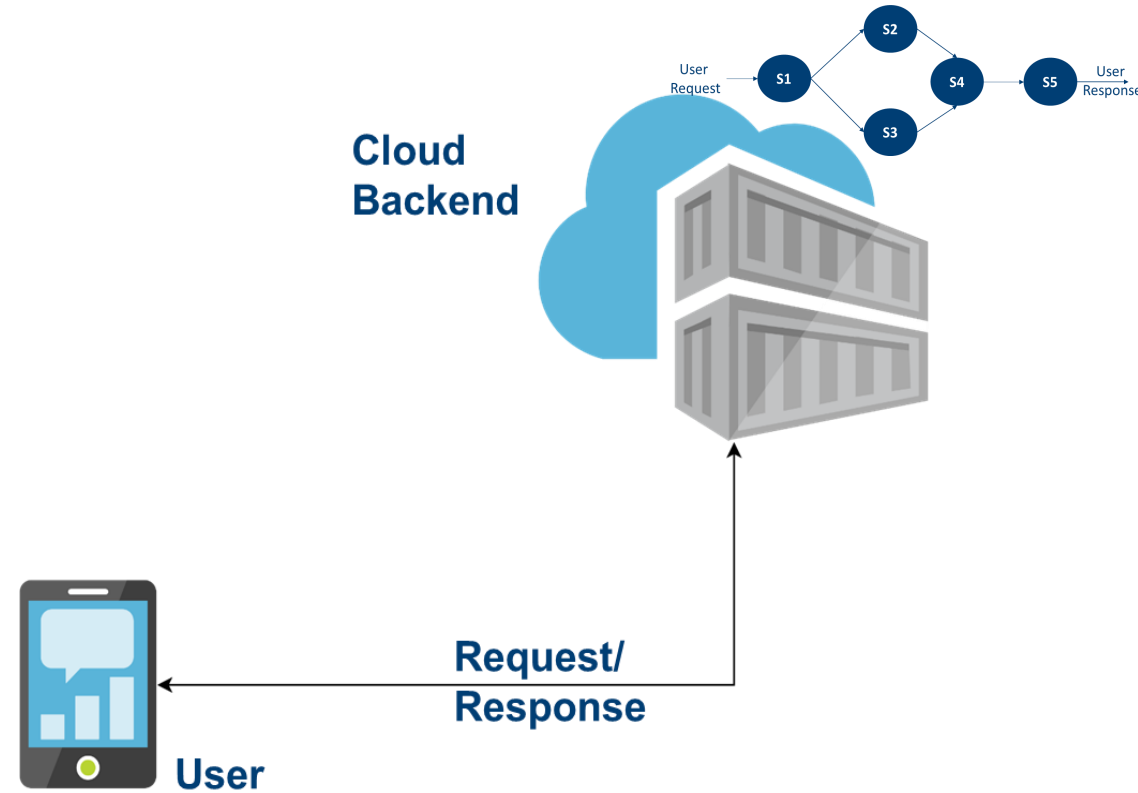


Data availability

Data traceability and monitoring

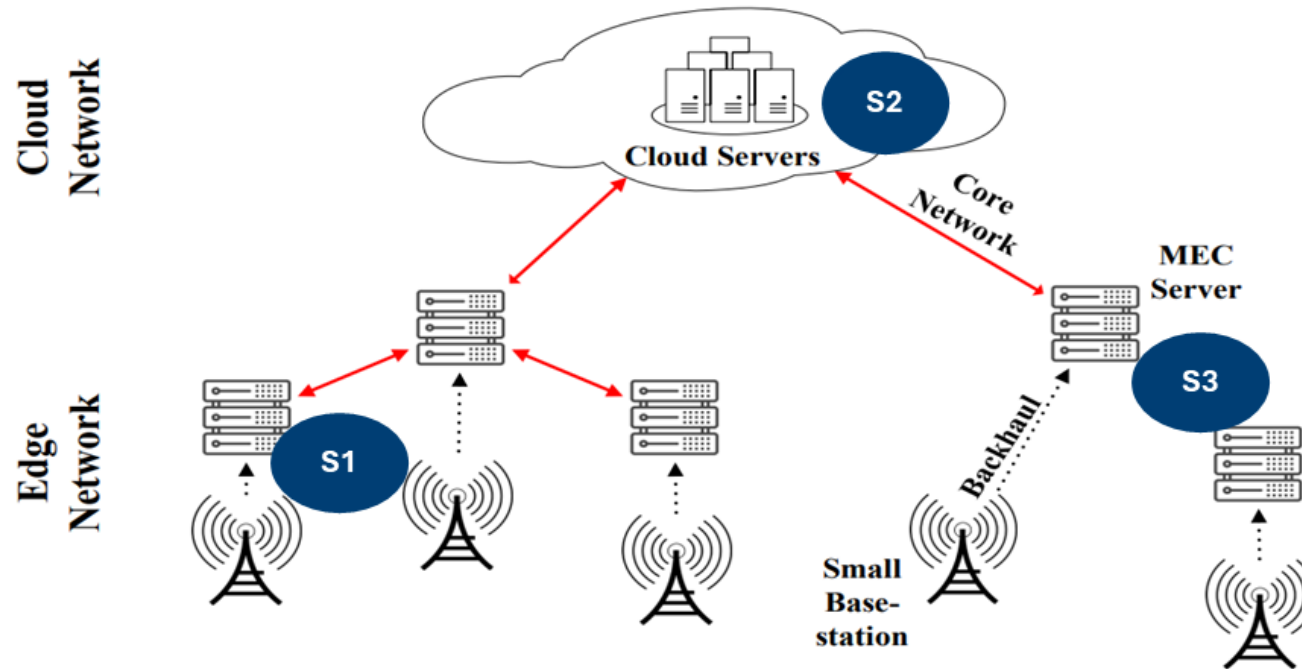
Data-Oriented Architectures

Prioritise decentralisation



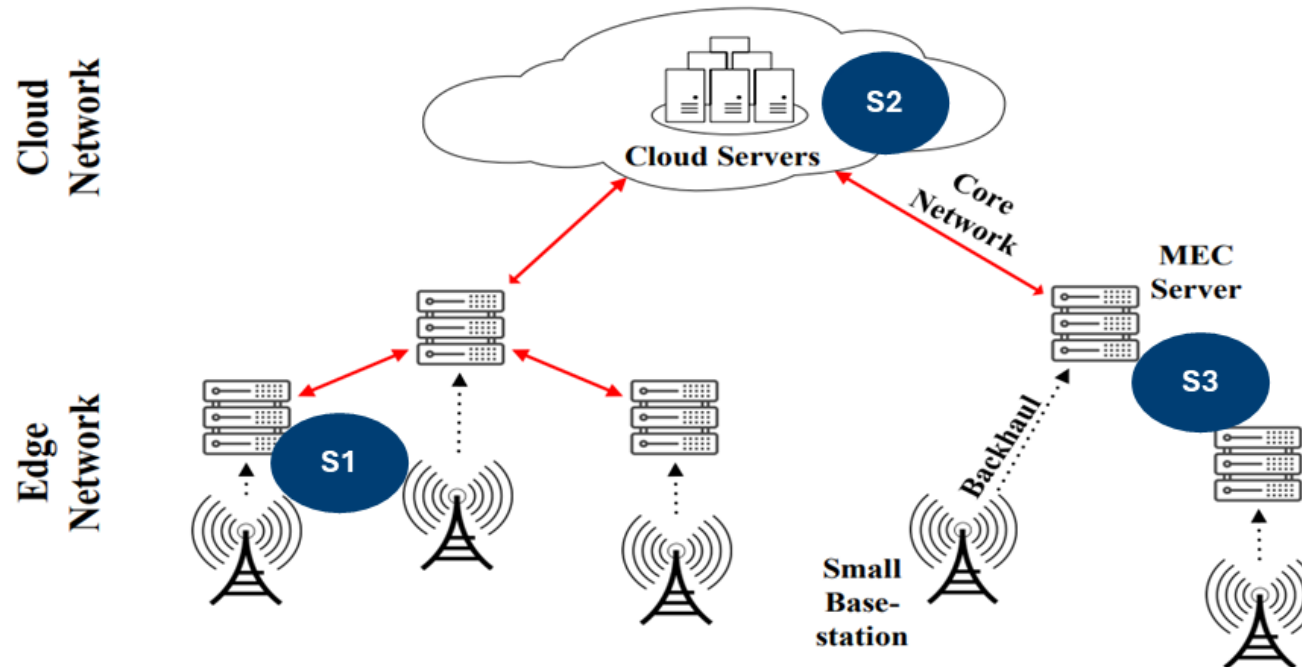
Data-Oriented Architectures

Prioritise decentralisation



Data-Oriented Architectures

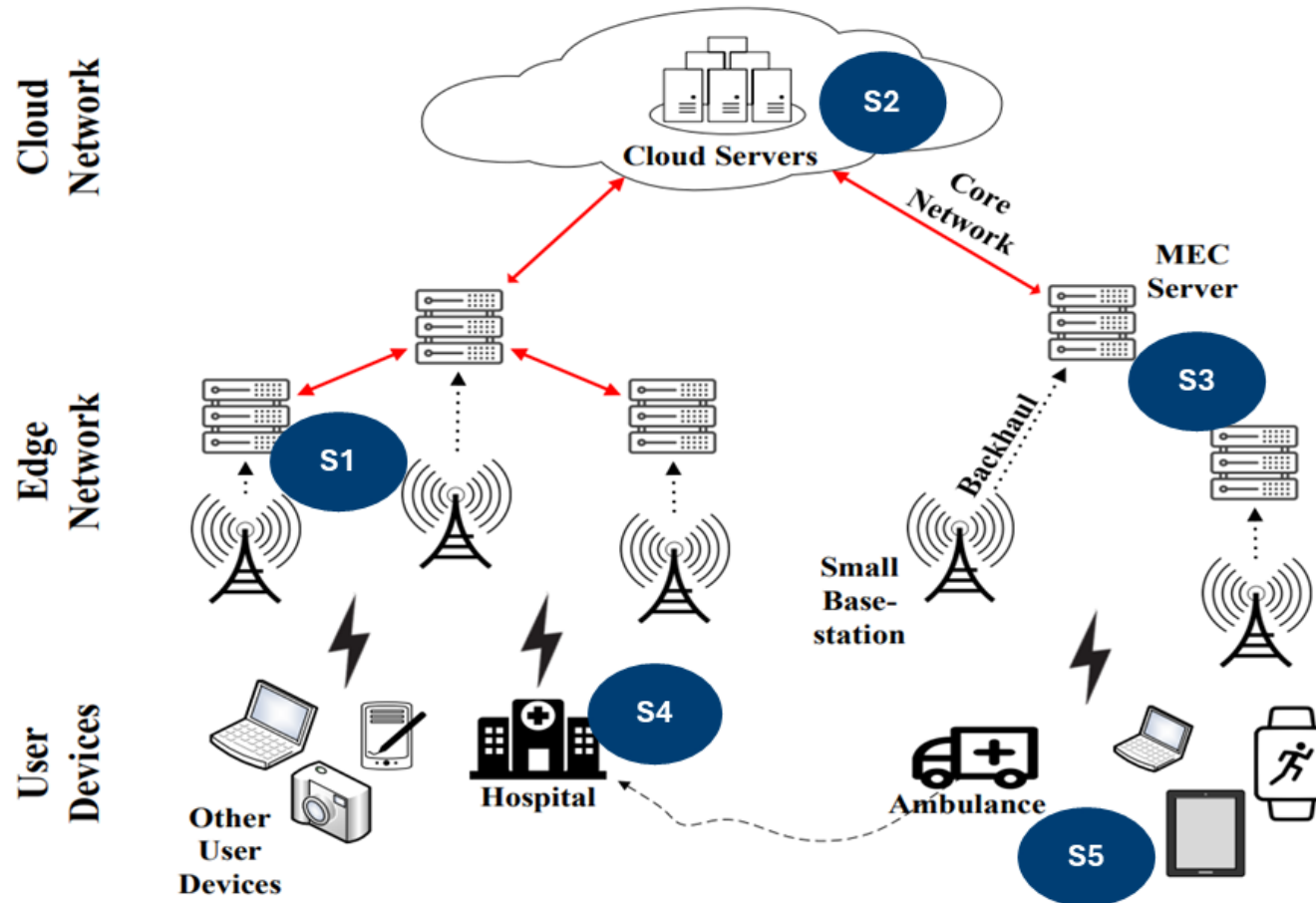
Prioritise decentralisation



Super-low latency requirements

Data-Oriented Architectures

Openness



Data ownership
Sustainability

Data-Oriented Architectures

 > cs > arXiv:2302.04810

Search...
Help | Advance

Computer Science > Software Engineering

[Submitted on 9 Feb 2023]

Real-world Machine Learning Systems: A survey from a Data-Oriented Architecture Perspective

Christian Cabrera, Andrei Paleyes, Pierre Thodoroff, Neil D. Lawrence

With the upsurge of interest in artificial intelligence machine learning (ML) algorithms, originally developed in academic environments, are now being deployed as parts of real-life systems that deal with large amounts of heterogeneous, dynamic, and high-dimensional data. Deployment of ML methods in real life is prone to challenges across the whole system life-cycle from data management to systems deployment, monitoring, and maintenance. Data-Oriented Architecture (DOA) is an emerging software engineering paradigm that has the potential to mitigate these challenges by proposing a set of principles to create data-driven, loosely coupled, decentralised, and open systems. However DOA as a concept is not widespread yet, and there is no common understanding of how it can be realised in practice. This review addresses that problem by contextualising the principles that underpin the DOA paradigm through the ML system challenges. We explore the extent to which current architectures of ML-based real-world systems have implemented the DOA principles. We also formulate open research challenges and directions for further development of the DOA paradigm.

Research work	Data as a First Class Citizen			Prioritise Decentralisation			Openness		
	Data driven	Invariant and shared data mode	Data coupling	Local data chunks	Local first	Peer-to-peer first	Autonomous entities	Asynchronous entities	Message exchange protocol
Junchen et al. [60]	✓	✓	✓	✓	✓	✓	✓	✓	✓
Lebofsky et al. [72]	✓	✓	✓	✓	✓	✓	✓	✓	✓
Herrero et al. [59]	✓	✓	✓	✓	✓	✓	✓	✓	✓
Zhang et al. [125]	✓	✓	✓	✓	✓	✓	✓	✓	✓
Karageorgou et al. [66]	✓	-	✓	✓	✓		✓	✓	✓
Sultana et al. [116]	✓	✓	✓	-			✓	✓	✓
Calancea et al. [29]	✓	✓		-	-		-	-	-
Schumann et al. [106]	✓	✓	✓				✓	✓	✓
Alves et al. [9]	✓	✓	✓				✓	✓	✓
De Caro et al. [36]	✓	✓	✓				✓	✓	✓
Nguyen et al. [83]	✓	✓	✓				-	✓	✓
Xu et al. [123]	✓	✓		✓	✓	✓	✓		
Alonso et al. [8]	✓	-		✓	-		✓		✓
Sarabia-Jácome et al. [103]	✓	-		✓	✓		-		-
Santana et al. [102]	✓	✓		-	-		-		✓
Shih et al. [112]	✓	✓	-				-	-	-
Lu et al. [75]	✓	✓		-		-	-		-
Brumbaugh et al. [22]	✓	-	-	-	-	-			
Shan et al. [109]	✓	✓		✓	✓	✓			
Schubert et al. [105]	✓	✓	✓					✓	✓
Dai et al. [33]	✓	✓	✓	✓	✓				
Zhang et al. [126]	✓	✓		✓	-		✓		
Quintero et al. [99]	✓			✓	✓		-		-
Habibi et al. [56]	✓	✓	-				-	-	-
Gorkin et al. [55]	✓			-			-	-	-
Shi et al. [111]	✓			✓	✓			✓	
Franklin et al. [47]	✓						✓	✓	✓
Bayerl et al. [15]	✓	-		✓	✓				
Bellocchio et al. [16]	✓						-	✓	✓
Johnny et al. [62]	✓				✓		-		-
Barachi et al. [14]	✓	✓		-			-		
Salhaoui et al. [101]	✓				-		-		-
Hegemier et al. [58]	✓				-		-		-
Cabanes et al. [23]	✓	✓	✓						
Agarwal et al. [2]	✓	✓	✓						
Müller et al. [81]	✓	✓	-						
Gao et al. [49]	✓	✓	-						
Amrollahi et al. [10]	✓	-	-						
Niu et al. [86]	✓			-			-		
Gallagher et al. [48]	✓			-			-		
Conroy et al. [31]	✓	✓					-		
Falcao et al. [43]	✓	✓							
Hawes et al. [57]	✓	✓							
Kemsaram et al. [67]	✓				✓				
Qiu et al. [98]	✓						-		
Ali et al. [7]	✓						-		

✓ = Adopted, - = Partially adopted, = Not adopted

Data-Oriented Architectures



[Submitted on 9 Feb 2023]

Real-world Machine Learning Systems: A survey from a Data-Oriented Architecture Perspective

Christian Cabrera, Andrei Paleyes, Pierre Thodoroff, Neil D. Lawrence

With the upsurge of interest in artificial intelligence machine learning (ML) algorithms, originally developed in academic environments, are now being deployed as parts of real-life systems that deal with large amounts of heterogeneous, dynamic, and high-dimensional data. Deployment of ML methods in real life is prone to challenges across the whole system life-cycle from data management to systems deployment, monitoring, and maintenance. Data-Oriented Architecture (DOA) is an emerging software engineering paradigm that has the potential to mitigate these challenges by proposing a set of principles to create data-driven, loosely coupled, decentralised, and open systems. However DOA as a concept is not widespread yet, and there is no common understanding of how it can be realised in practice. This review addresses that problem by contextualising the principles that underpin the DOA paradigm through the ML system challenges. We explore the extent to which current architectures of ML-based real-world systems have implemented the DOA principles. We also formulate open research challenges and directions for further development of the DOA paradigm.

Few projects fully follow DOA principles.

Most of the solutions are centralised and cloud-based.

Databases, streams and message queues enable the data first principle.

Distributed storage and computing technologies for decentralisation.

Asynchronous communication for openness.

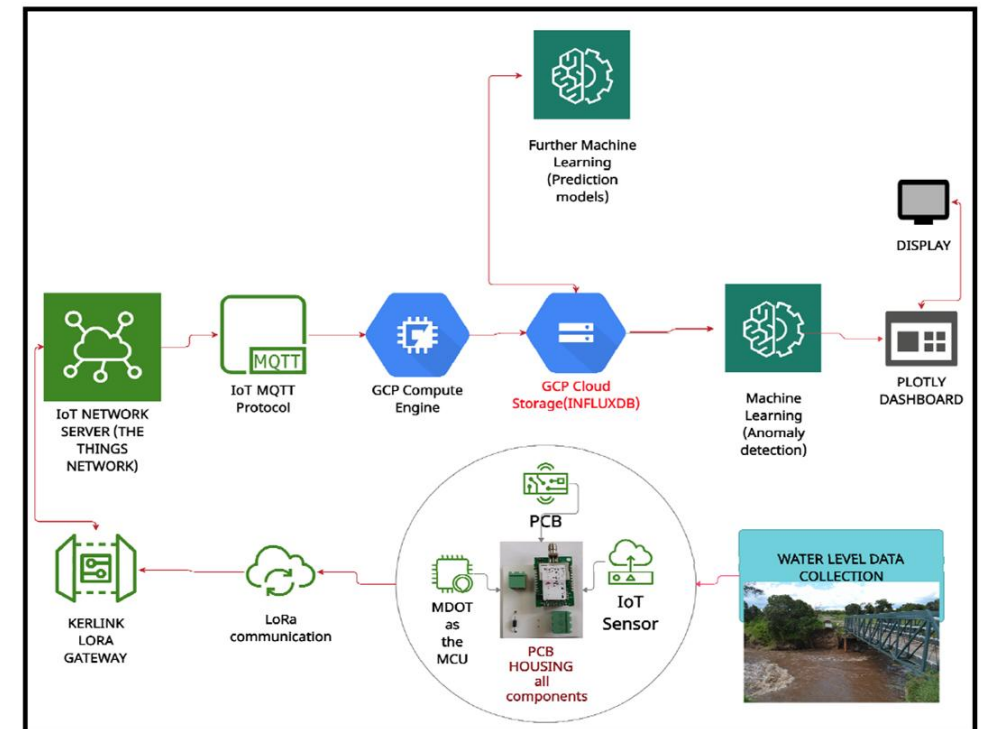
Data-Oriented Architectures



Source: https://en.wikipedia.org/wiki/Ewaso_Ng%27iro

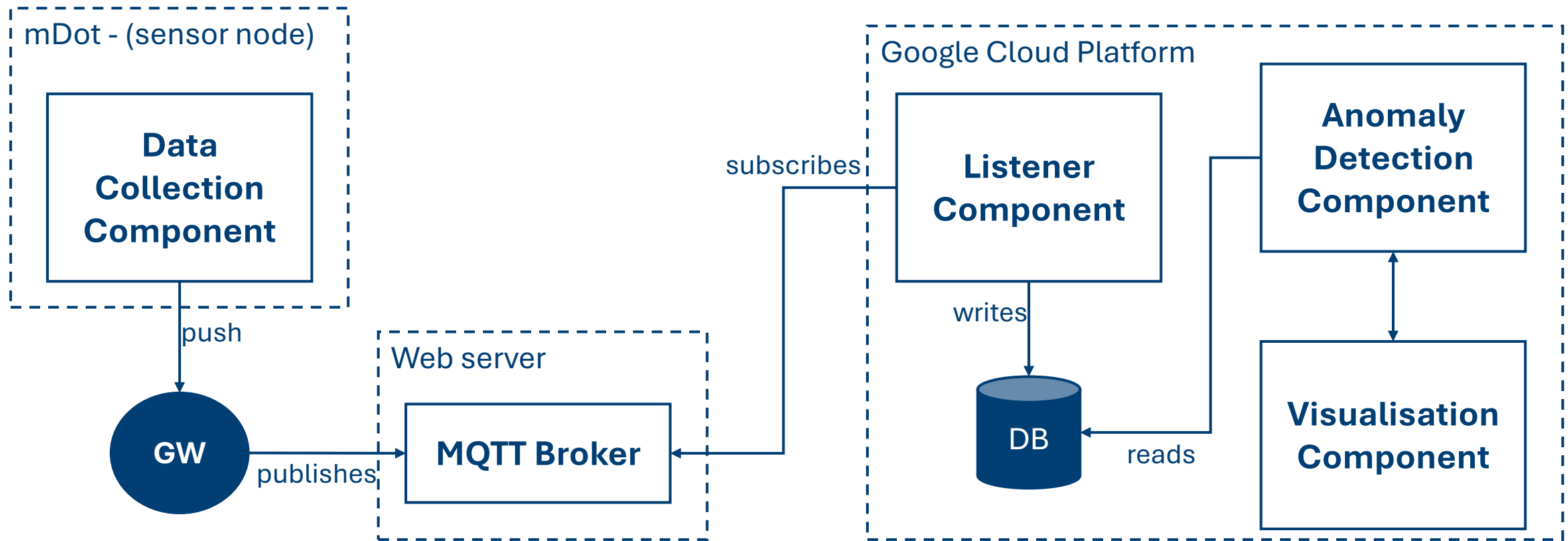
Ewaso Nyiro River - Kenya

Water level monitoring project at DeKUT [1]



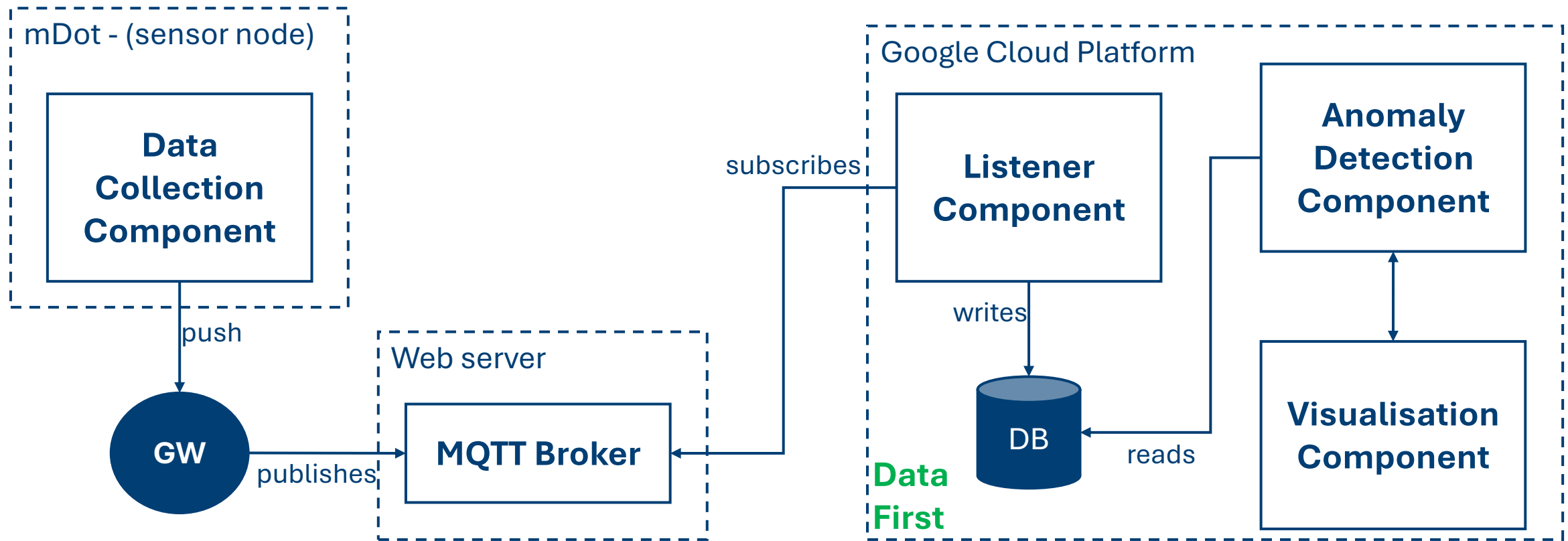
Data-Oriented Architectures

Water level monitoring project at DeKUT [1]



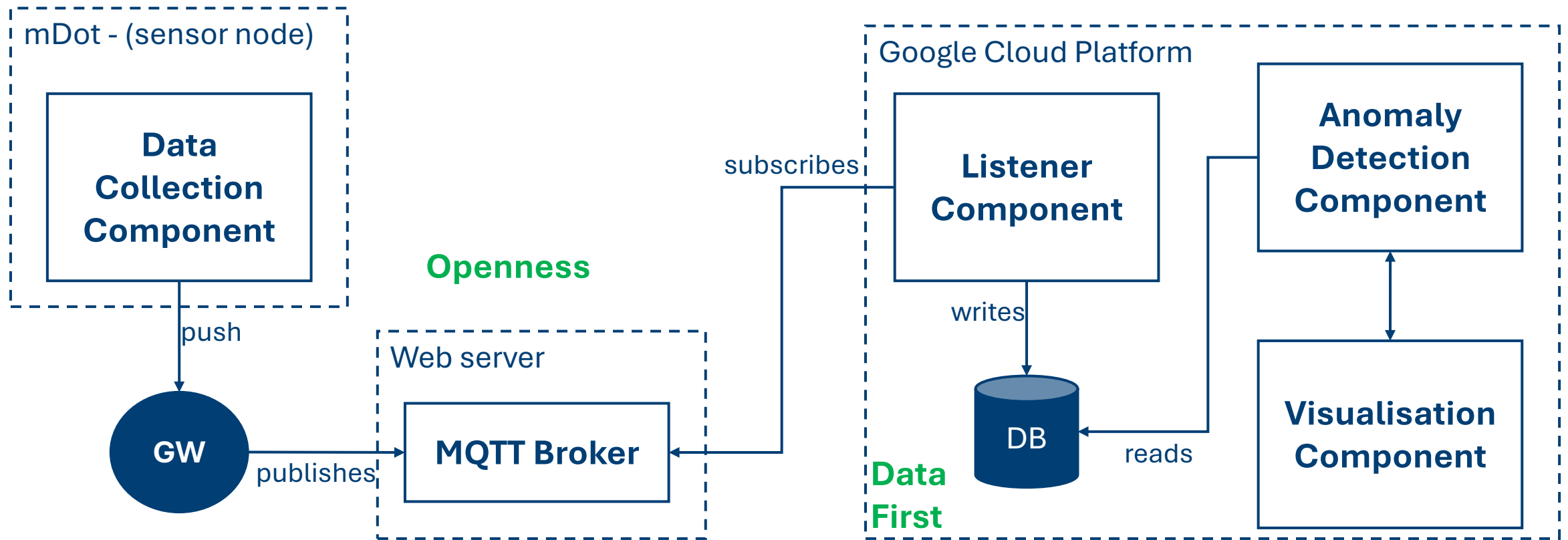
Data-Oriented Architectures

Water level monitoring project at DeKUT [1]



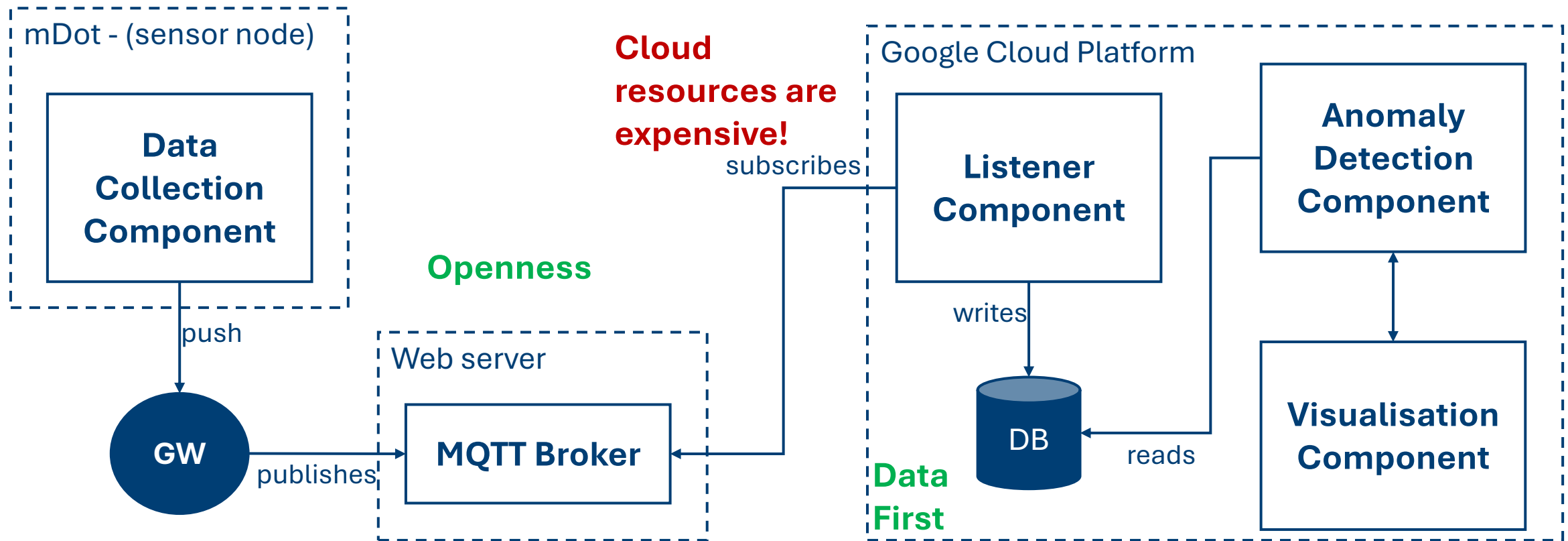
Data-Oriented Architectures

Water level monitoring project at DeKUT [1]



Data-Oriented Architectures

Water level monitoring project at DeKUT [1]



Data-Oriented Architectures

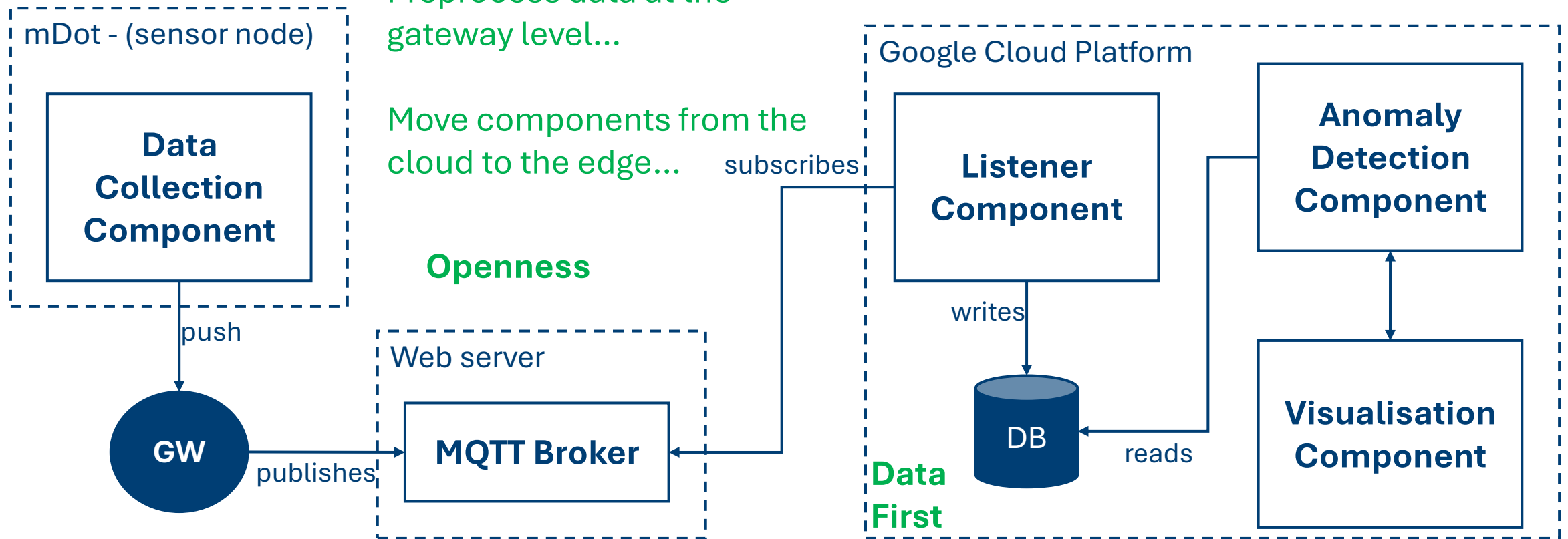
Decentralisation

Preprocess data at the gateway level...

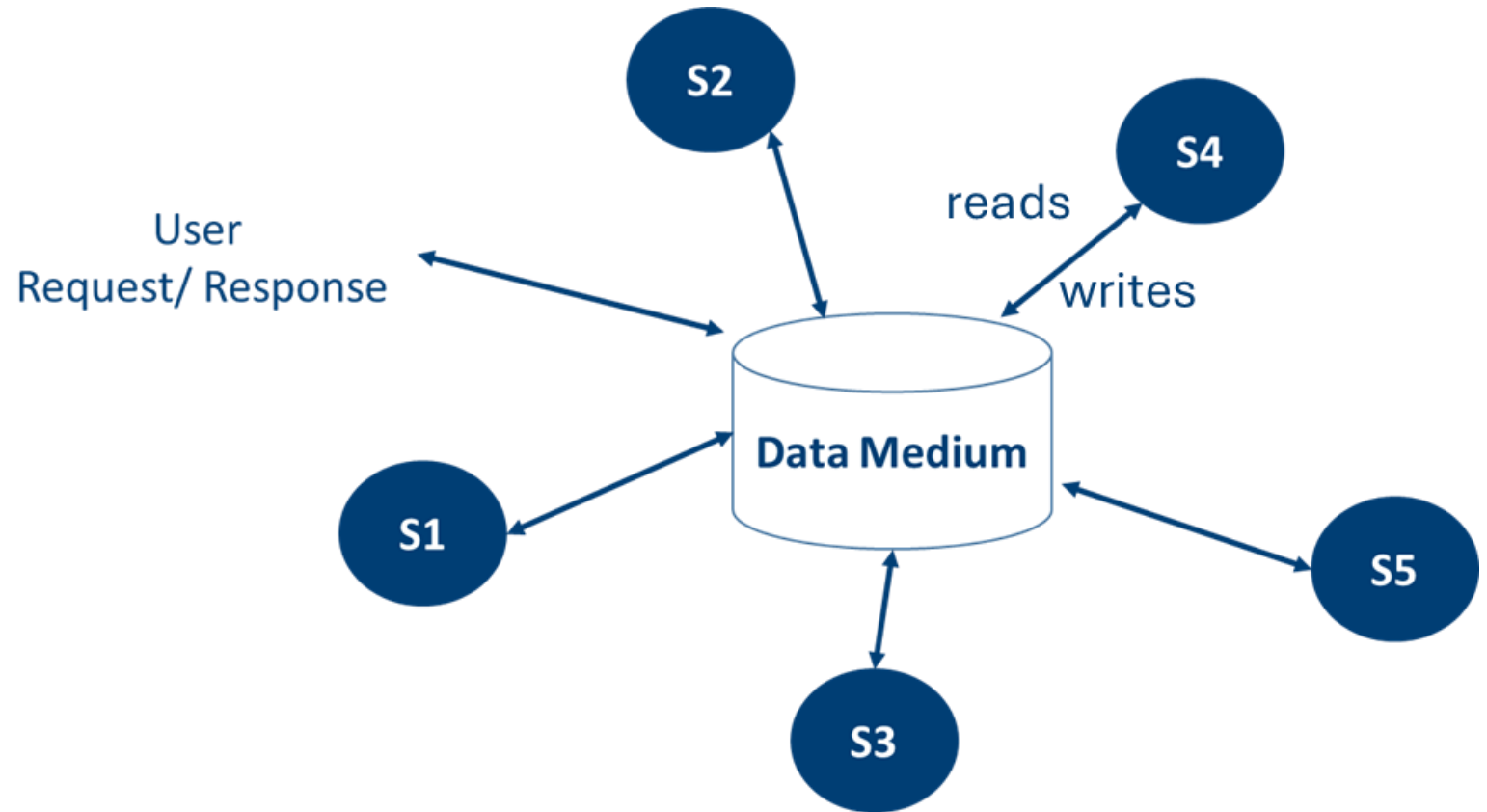
Move components from the cloud to the edge...

Openness

Water level monitoring project at DeKUT [1]



Data-First Principle



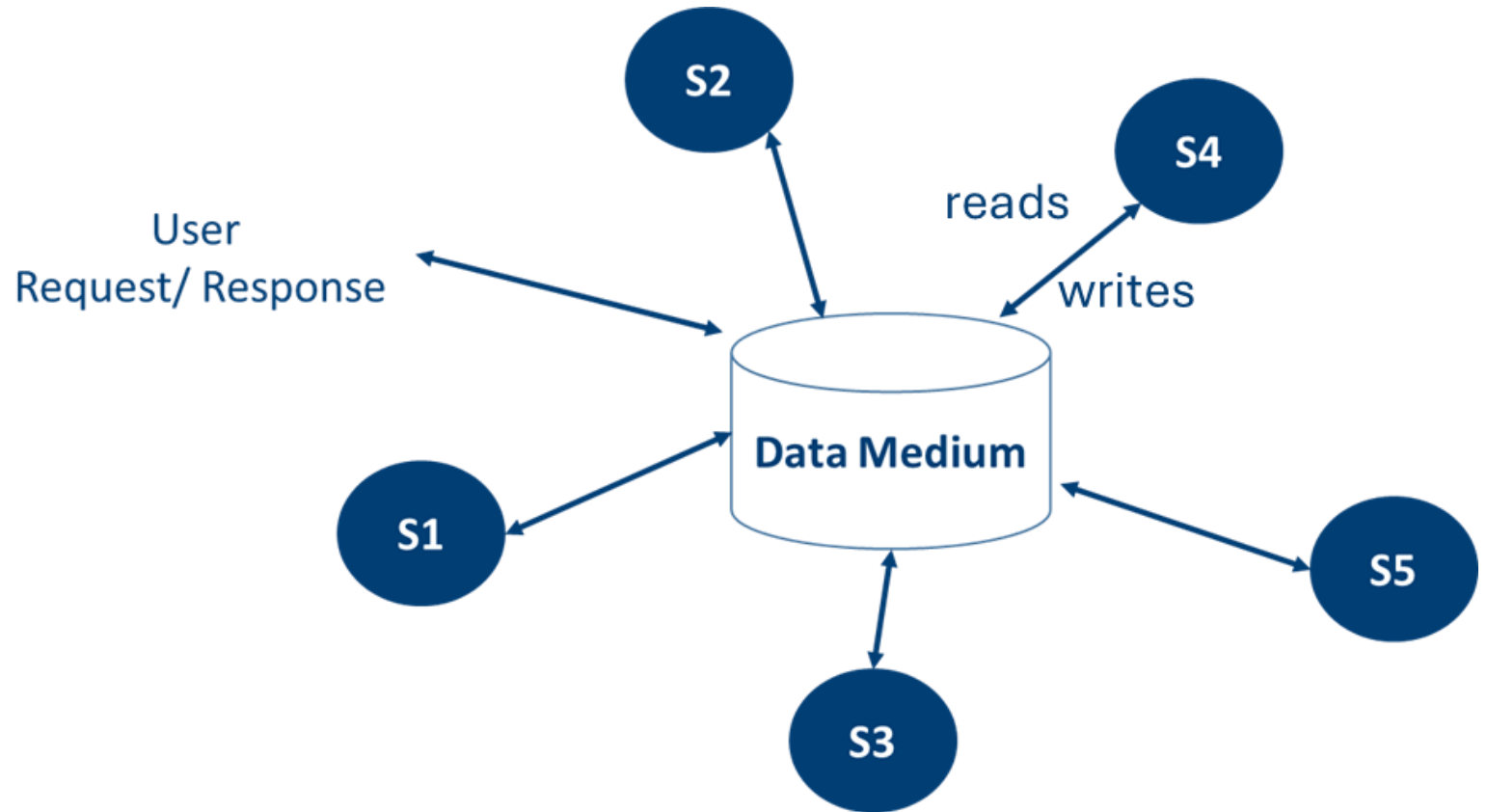
Data-First Principle

Facilitates data collection processes.

Asynchronous communication fits better when manipulating large data sets.

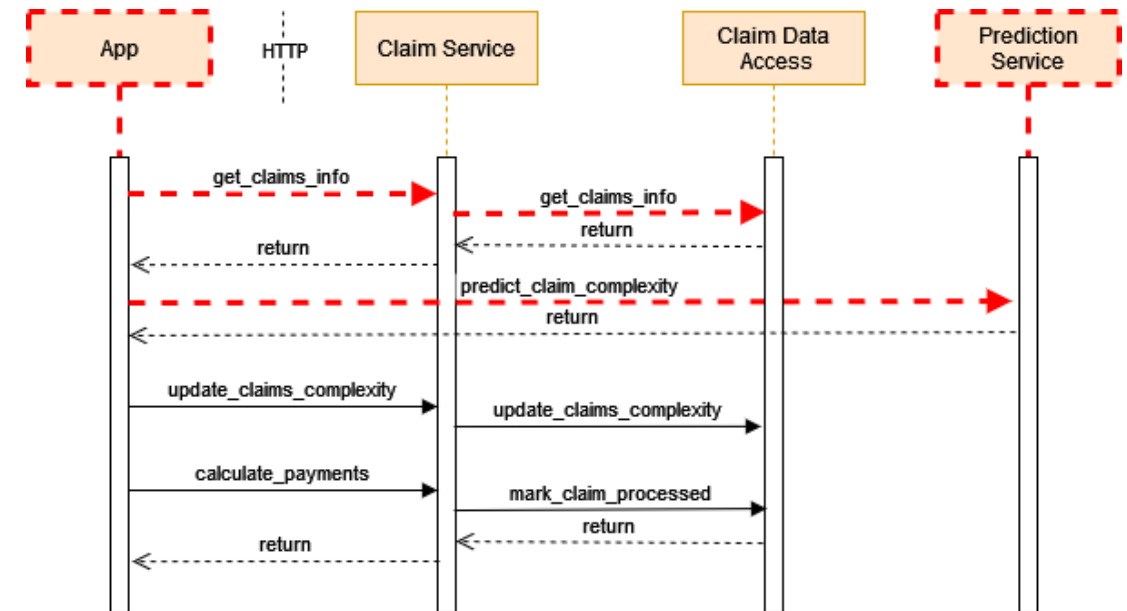
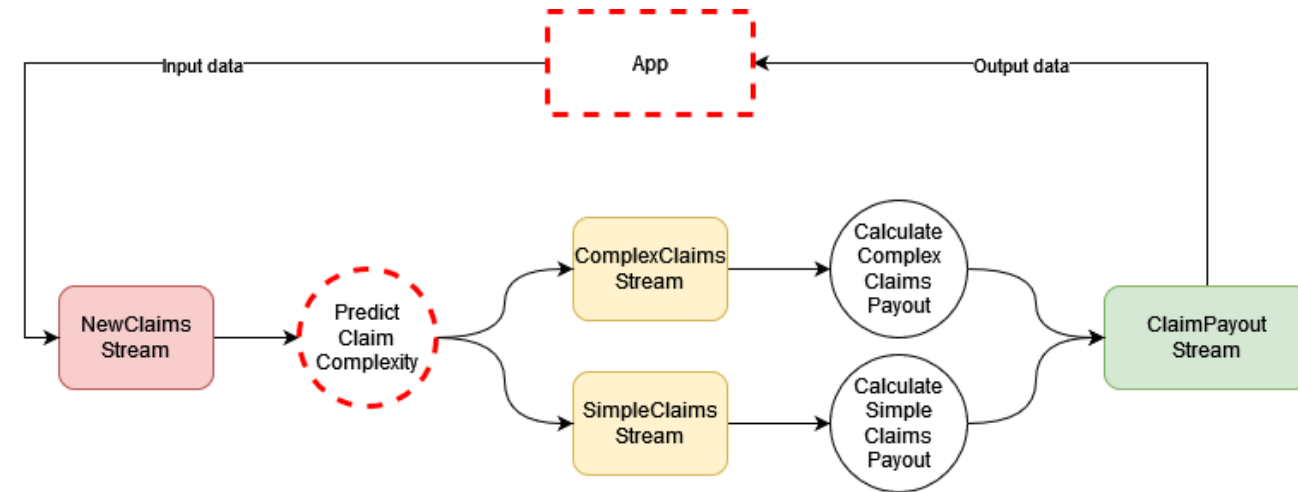
The Data Medium is a shared model that facilitates monitoring tasks.

The historical state of the systems is available by design.



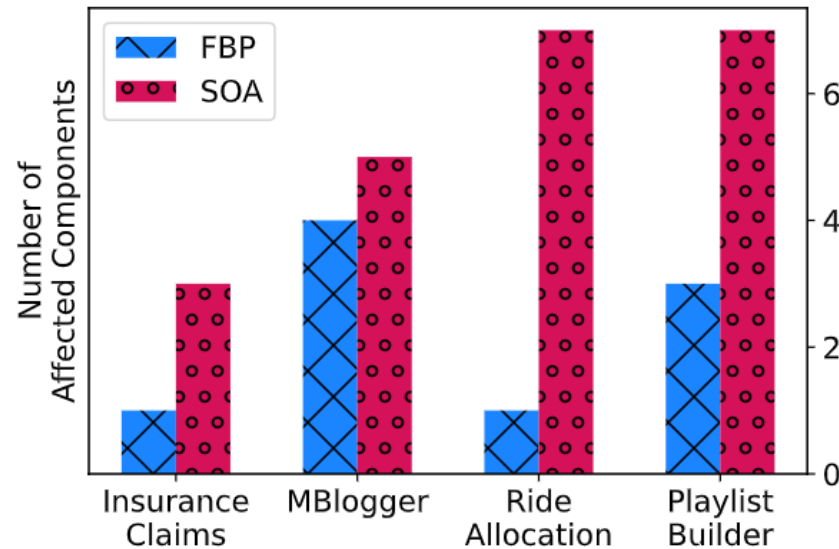
Data-First Principle

DOA vs SOA

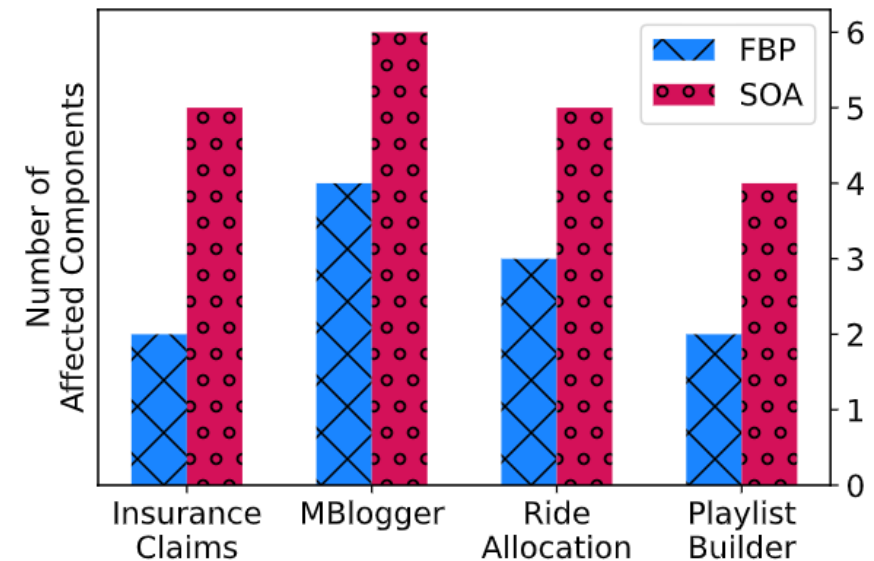


Data-First Principle

DOA vs SOA – Number of Affected Components



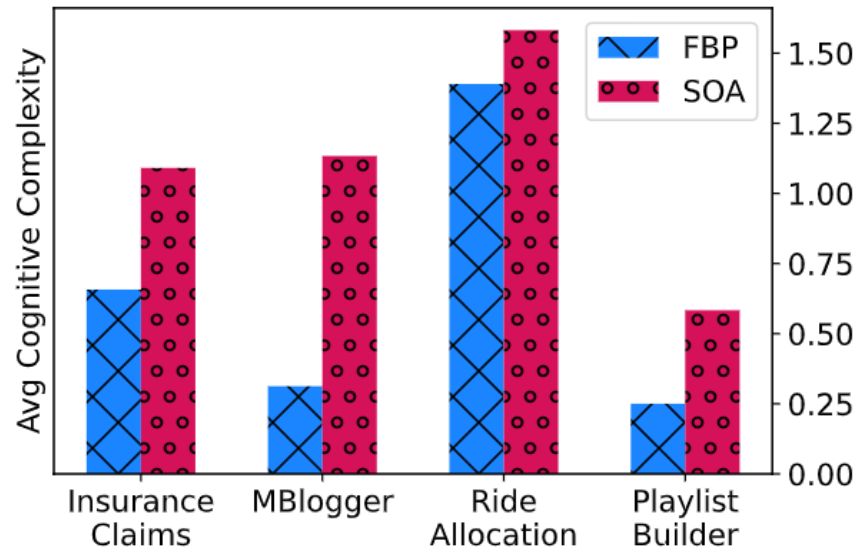
From Baseline to Data Collection



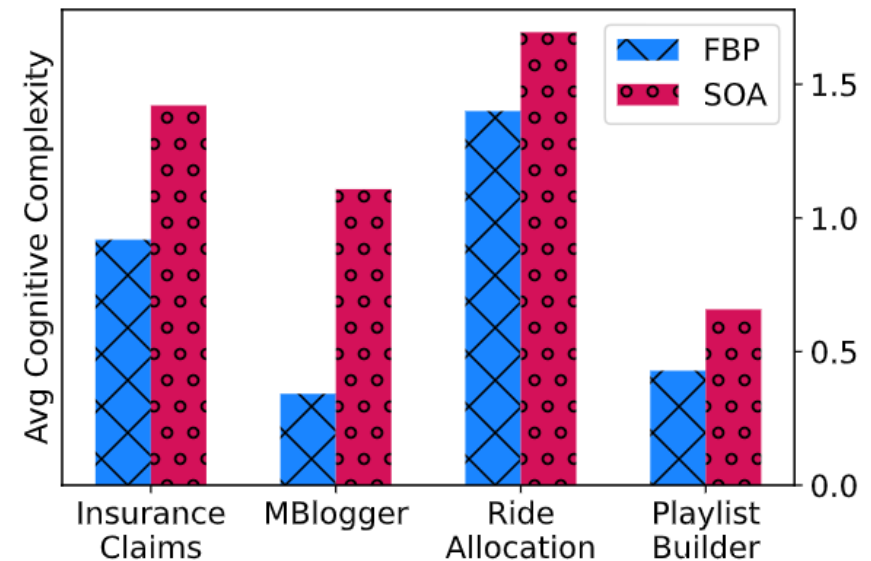
From Data Collection to ML

Data-First Principle

DOA vs SOA – Cognitive Complexity



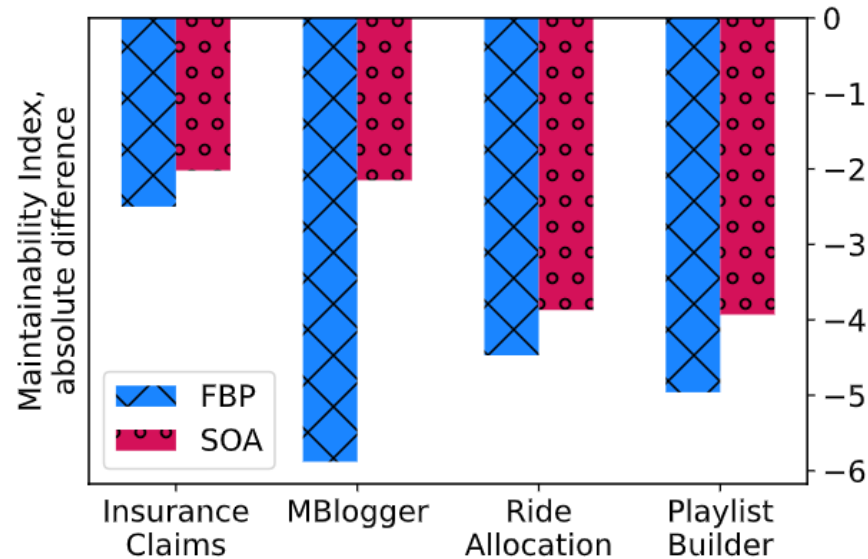
From Baseline to Data Collection



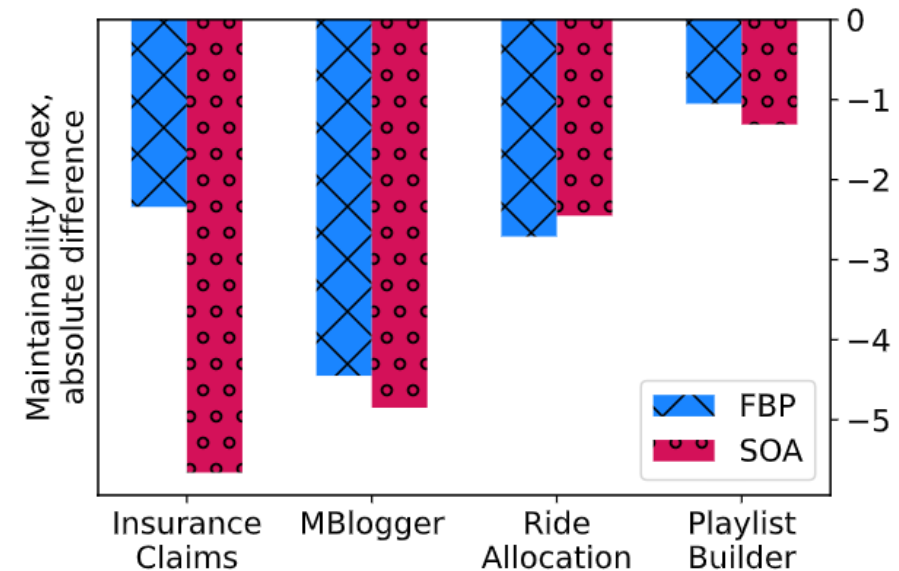
From Data Collection to ML

Data-First Principle

DOA vs SOA – Maintainability Index



From Baseline to Data Collection



From Data Collection to ML

Data-First Principle

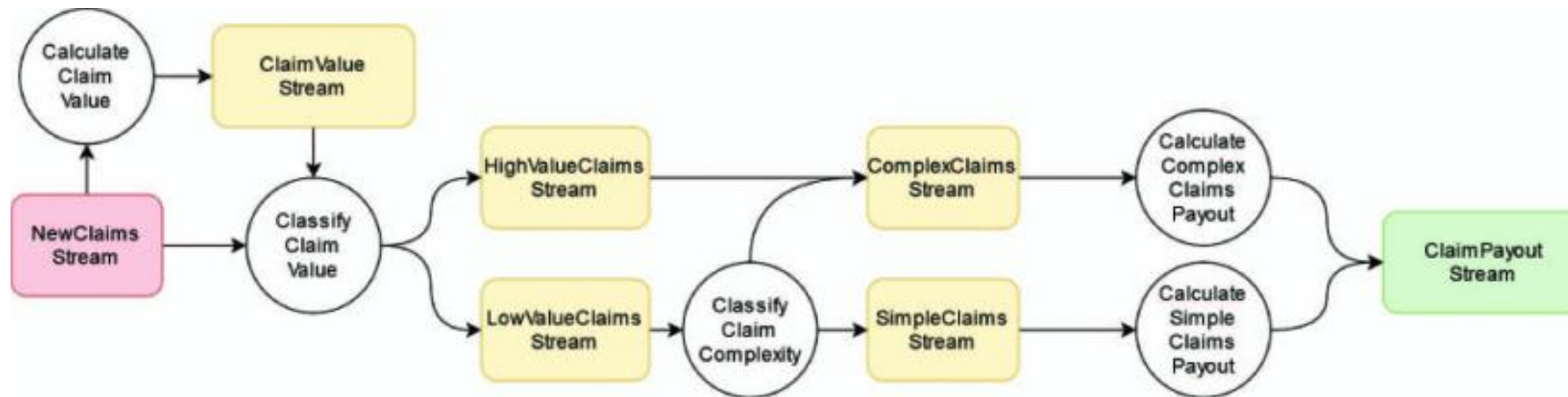
DOA and Causality Analysis

Dataflow graphs as complete causal graphs

Andrei Paleyes^{*1}, Siyuan Guo^{*12}, Bernhard Schölkopf², Neil D. Lawrence¹

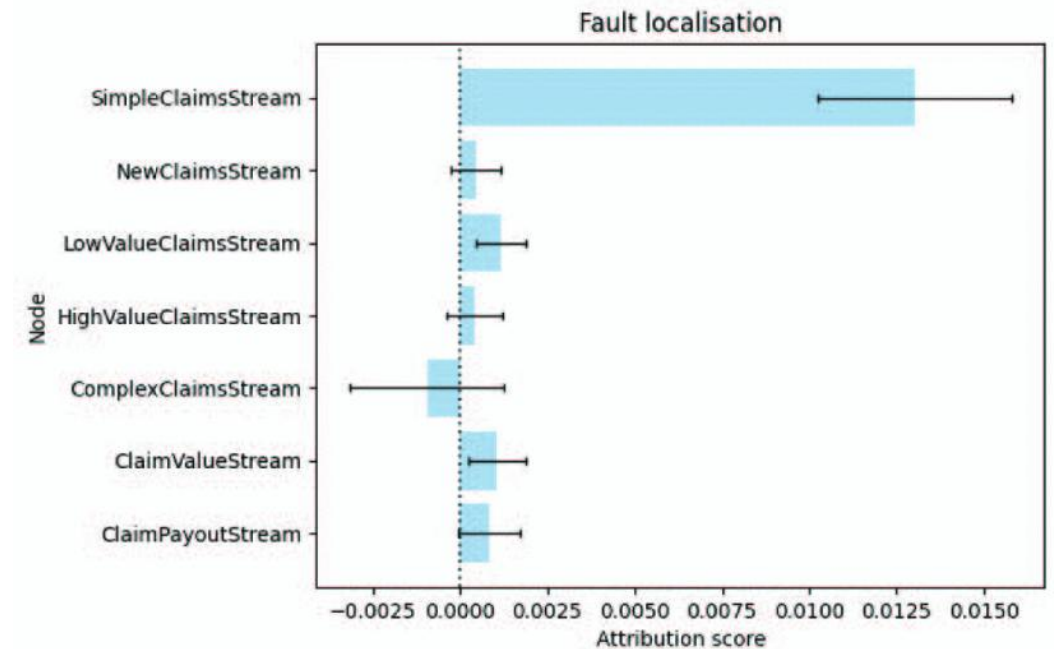
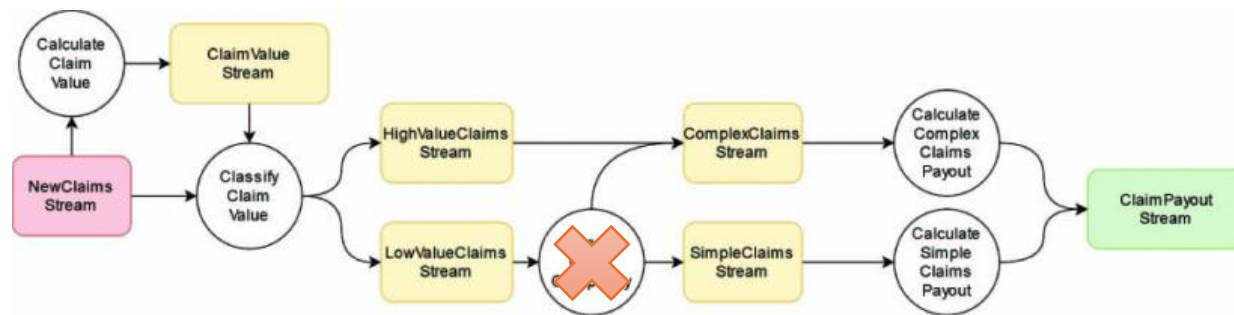
¹Department of Computer Science and Technology, University of Cambridge

²Max Planck Institute for Intelligent Systems



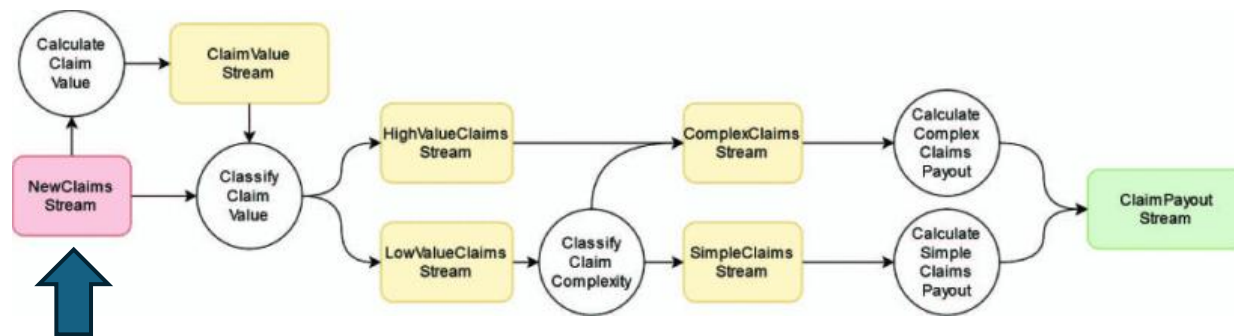
Data-First Principle

DOA and Causality Analysis – Fault localisation

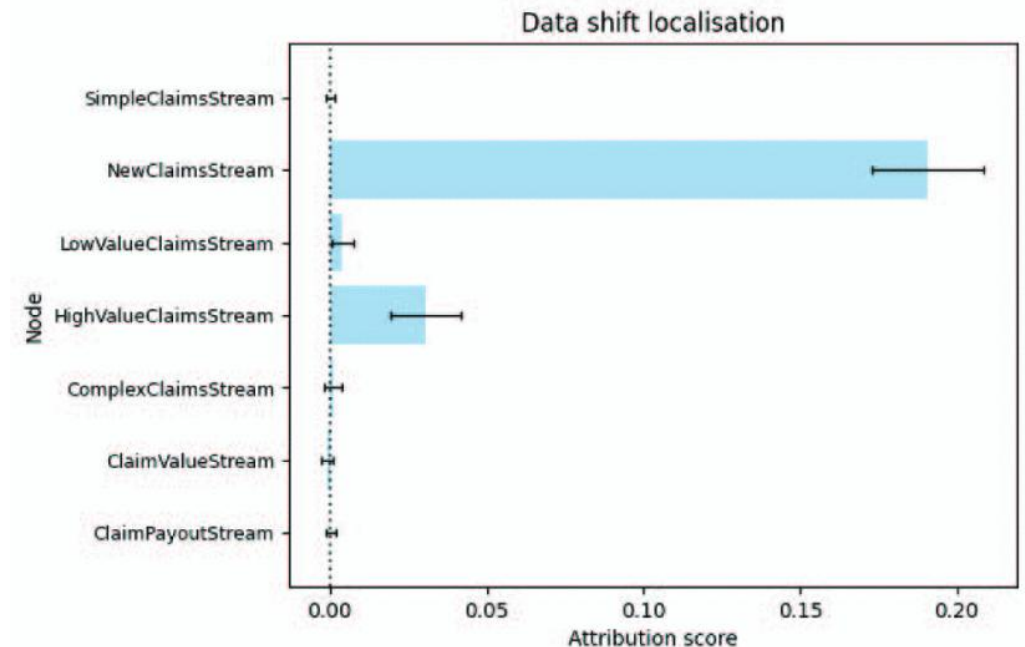


Data-First Principle

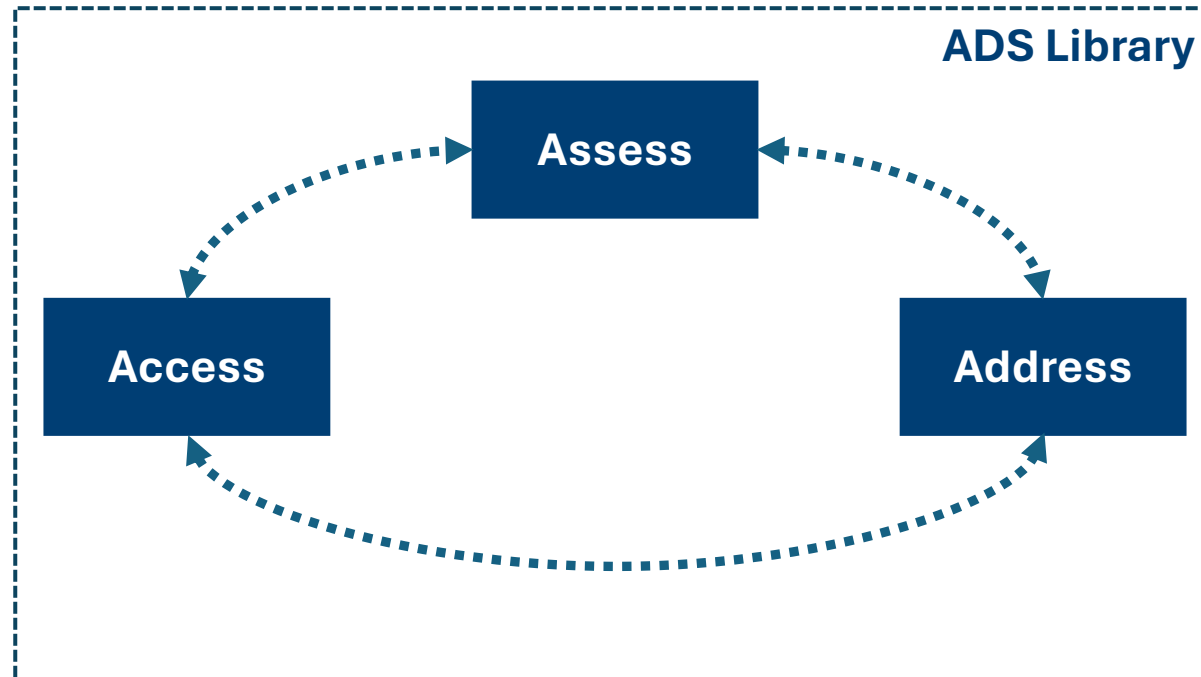
DOA and Causality Analysis – Identifying data shifts



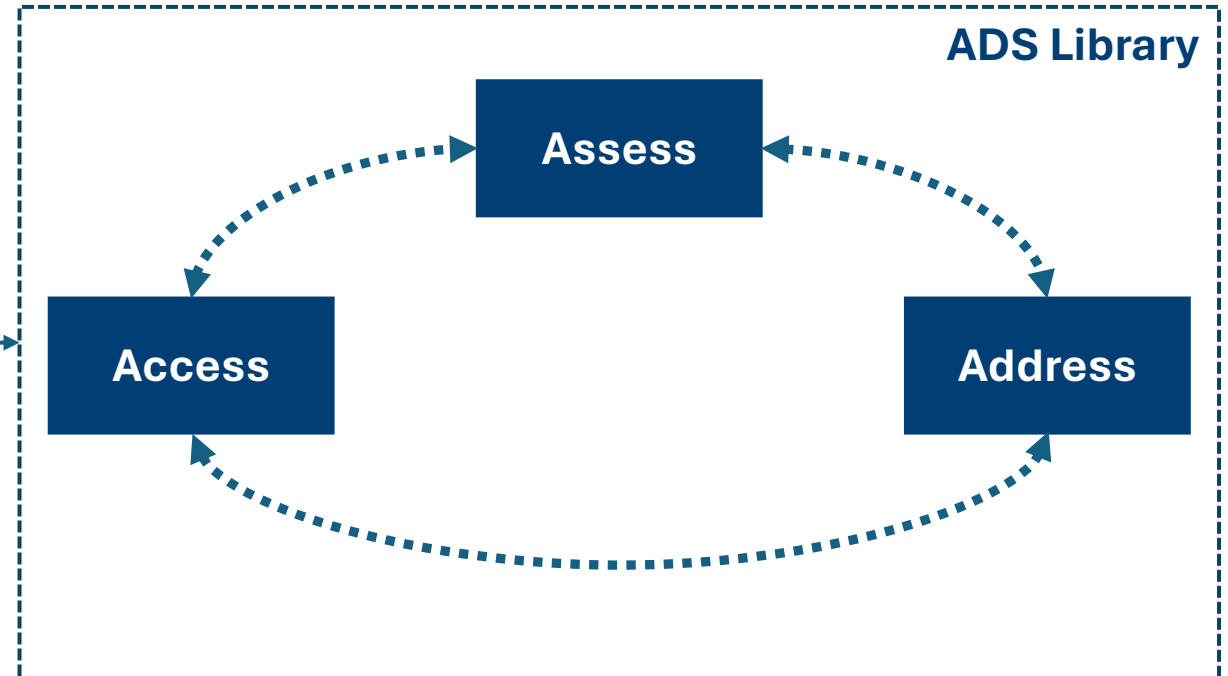
50% up on the
originally claimed
amount.



Data-First Principle



Data-First Principle



Summary

- Systems design decisions change between systems, but these usually share requirements.
- Similar requirements can be addressed following similar solutions.
- Data-driven systems demand to design systems that prioritise data.
- Data-Oriented Architectures is a useful paradigm to design data-driven systems.
- The Data-First principle is particularly relevant for our data science pipelines.

Many thanks!