Bidirectional Data Pipelines: An Industrial Case Study

Aiswarya Raj Munappy[1], Anas Dakkak[2], Jan Bosch[1], and Helena Olsson[3]

[1]Chalmers tekniska hogskola AB
[2]Telefonaktiebolaget LM Ericsson
[3]Malmo universitet Institutionen for datavetenskap och medieteknik

April 09, 2024

Abstract

Background: Bidirectional data pipelines have emerged as a response to the evolving needs of modern data ecosystems. Traditionally, unidirectional pipelines allowed data to flow in a single direction, limiting interaction. The surge in demand for real-time bidirectional communication prompted the development of pipelines that enable two-way data flow, facilitating seamless and dynamic exchanges between source and destination. *Objective*: The research aims to delve into the role of bidirectional data pipelines within the companies producing and selling software-intensive embedded systems products. Further, the study endeavors to elucidate the fundamental differences between unidirectional and bidirectional data pipelines, shedding light on their unique characteristics. Through comprehensive exploration, it seeks to discern the benefits and challenges inherent in implementing and maintaining bidirectional data pipelines. Furthermore, a critical aspect of the research involves outlining the intricate steps and considerations essential for migrating from unidirectional to bidirectional data pipelines. This includes a focus on prerequisites, methodologies, and the potential benefits derived from such a transition. *Method*: This study employs a qualitative research approach centered around a multiple-interpretive case study to delve into the complexities of bidirectional data pipelines. Five distinct use cases have been meticulously selected to provide a comprehensive understanding of various aspects of bidirectional data pipelines. Through the in-depth analysis of these concrete use cases, this research aims to elucidate the intricacies, benefits, and challenges associated with bidirectional data pipelines in software-intensive embedded systems environment. *Results*: The study yielded insightful results on various aspects of bidirectional data pipelines, emphasizing their distinctions from unidirectional data pipelines without a shared data transmission channel. It uncovered the compelling need for bidirectional data pipelines in modern data-centric environments, where the dynamic exchange of information between source and destination is pivotal. The identified benefits ranged from enhanced real-time data synchronization to improved responsiveness in addressing evolving business requirements. Concurrently, the study elucidated inherent challenges, such as increased complexity in pipeline management and potential security considerations. Moreover, the research provided a nuanced understanding of the stepwise process involved in transitioning from unidirectional to bidirectional data pipelines. *Conclusion*: The study underscores the pivotal role of bidirectional data pipelines in meeting evolving data requirements, offering benefits like real-time synchronization and improved decision-making. Despite their importance, there is a lack of comprehensive research on bidirectional pipelines, prompting the need for further exploration. The transitional journey involves addressing challenges such as data consistency and security concerns while optimizing system design for compatibility. For researchers, the study suggests avenues for optimizing bidirectional pipeline performance, examining long-term impacts, scalability, and real-time anomaly detection. Practitioners can leverage insights for informed decision-making in transitioning to bidirectional data pipelines, aligning organizational needs with their benefits and challenges.

Bidirectional Data Pipelines: An Industrial Case Study

Aiswarya Raj Munappy[a,], Anas Dakkak[c], Jan Bosch[a], Helena Holmström Olsson[b]

[a]*Department of Computer Science and Engineering, Chalmers University of Technology, Hörselgången 11, 412 96 Gothenburg, Sweden*
[b]*Department of Computer Science and Media Technology, Malmö University, Nordenskiöldsgatan, 205 06 Malmö, Sweden*
[c]*Ericsson, Torshamnsgatan 21, 164 40 Kista, Stockholm, Sweden*

Abstract

Background: Bidirectional data pipelines have emerged as a response to the evolving needs of modern data ecosystems. Traditionally, unidirectional pipelines allowed data to flow in a single direction, limiting interaction. The surge in demand for real-time bidirectional communication prompted the development of pipelines that enable two-way data flow, facilitating seamless and dynamic exchanges between source and destination.
Objective: The research aims to delve into the role of bidirectional data pipelines within the companies producing and selling software-intensive embedded systems products. Further, the study endeavors to elucidate the fundamental differences between unidirectional and bidirectional data pipelines, shedding light on their unique characteristics. Through comprehensive exploration, it seeks to discern the benefits and challenges inherent in implementing and maintaining bidirectional data pipelines. Furthermore, a critical aspect of the research involves outlining the intricate steps and considerations essential for migrating from unidirectional to bidirectional data pipelines. This includes a focus on prerequisites, methodologies, and the potential benefits derived from such a transition.
Method: This study employs a qualitative research approach centered around a multiple-interpretive case study to delve into the complexities of bidirectional data pipelines. Five distinct use cases have been meticulously selected to provide a comprehensive understanding of various aspects of bidirectional data pipelines. Through the in-depth analysis of these concrete use cases, this research aims to elucidate the intricacies, benefits, and challenges associated with bidirectional data pipelines in software-intensive embedded systems environment.
Results: The study yielded insightful results on various aspects of bidirectional data pipelines, emphasizing their distinctions from unidirectional data pipelines without a shared data transmission channel. It uncovered the compelling need for bidirectional data pipelines in modern data-centric environments, where the dynamic exchange of information between source and destination is pivotal. The identified benefits ranged from enhanced real-time data synchronization to improved responsiveness in addressing evolving business requirements. Concurrently, the study elucidated inherent challenges, such as increased complexity in pipeline management and potential security considerations. Moreover, the research provided a nuanced understanding of the stepwise process involved in transitioning from unidirectional to bidirectional data pipelines.
Conclusion: The study underscores the pivotal role of bidirectional data pipelines in meeting evolving data requirements, offering benefits like real-time synchronization and improved decision-making. Despite their importance, there is a lack of comprehensive research on bidirectional pipelines, prompting the need for further exploration. The transitional journey involves addressing challenges such as data consistency and security concerns while optimizing system design for compatibility. For researchers, the study suggests avenues for optimizing bidirectional pipeline performance, examining long-term impacts, scalability, and real-time anomaly detection. Practitioners can leverage insights for informed decision-making in transitioning to bidirectional data pipelines, aligning organizational needs with their benefits and challenges.

Keywords: Bidirectional data pipelines, significance, challenges, benefits, migration

1. Introduction

Data has become an invaluable asset in the era of artificial intelligence (AI), as it serves as a raw material for the development, training, and retraining of AI models [1]. The exponential increase in digital information has fueled the advancement of AI [2]. In fact, data has now become a necessity for the development of intelligent systems [3]. Therefore, responsible, efficient, and strategic management of data becomes essential for enabling artificial intelligence to reach its full potential across a range of domains as the field evolves [4]. These kinds of data-rich environments strongly demand data pipelines for better management of the data [5]. Consequently, data pipelines are becoming increasingly important for organizations to manage their data and turn it into insights and actionable information [6].

Data pipelines facilitate the integration of data from diverse sources, including internal systems, external databases, and third-party applications [7]. By streamlining the process of collecting and consolidating data, data pipelines enable organizations to create a comprehensive and unified view of their operations and customers [8]. In today's fast-changing world, the ability to process and analyze data in real-time is paramount. Data pipelines enable the seamless flow of data, allowing organizations to make timely, data-driven decisions, respond quickly to market changes, and capitalize on emerging opportunities [9].

Furthermore, data pipelines help ensure the quality and consistency of data by automating data cleansing, transformation, and validation processes [7, 10]. By standardizing and cleaning data as it moves through the pipeline, organizations can rely on accurate, reliable, and high-quality data for business analysis and decision-making [11]. By automating data transfer and processing tasks, pipelines reduce manual work, minimize errors, and enhance overall operational efficiency, allowing employees to focus on high-value tasks that drive business growth [12]. Therefore, data pipelines serve as the foundation for effective business intelligence and analytics initiatives. By delivering timely and accurate data to analytics tools and platforms, pipelines enable organizations to uncover valuable insights, identify trends, and make data-driven predictions, leading to informed business strategies and improved performance [5, 12].

Data pipelines are often considered one-directional, as data flows from data sources to the receiving destination. However, with the growing demand for real-time data synchronization, seamless communication across diverse environments, and integrity in an increasingly interconnected and dynamic digital landscape, two-way data flow becomes essential [13]. In addition, bidirectional data pipelines open a new horizon of possibilities for companies as they enable closed-loop automation. For example, bidirectional data pipelines enable companies to provide preemptive and proactive customer support [14]. Similarly, bidirectional data pipelines enable healthcare providers to send patient feedback and intervene when necessary during the patient intersession process used with psychology and psychotherapy treatments [15].

However, while bidirectional data pipelines can enable significant opportunities, the existing literature on bidirectional data pipelines is scarce. Up to the authors' knowledge, there is a limited body of research focusing on bidirectional data pipelines and exploring their specific characteristics. Therefore, this study aims to provide a comprehensive examination of bidirectional data pipelines in the software-intensive embedded systems domain by addressing the following objectives:

1. Highlight the key differences between unidirectional and bidirectional data pipelines.
2. Explain the significance of bidirectional data pipelines in a modern data-centric environment.
3. Identify the benefits and challenges associated with the implementation and management of bidirectional data pipelines.
4. Outline a roadmap for the smooth migration from unidirectional to bidirectional data pipelines, minimizing disruptions and maximizing the benefits.

The contribution of this paper is threefold. First, through a comprehensive comparison of unidirectional and bidirectional data pipelines, the study highlights the key differences, strengths, and limitations of each type, enabling organizations to make informed decisions regarding the selection of the most suitable data pipeline architecture based on their specific data management requirements and business objectives. Second, we identify the benefits and challenges associated with

implementing and managing bidirectional data pipelines using an industrial multiple-case study conducted at a multinational telecommunications vendor. Third, we outline the roadmap for a smooth and successful migration process from unidirectional to bidirectional data pipelines that minimize disruptions and maximize the benefits of bidirectional data synchronization.

The remainder of this paper is organized as follows: The background is presented next in Section 2, followed by the description of the research method in Section 3. The further section presents a comprehensive understanding of bidirectional data pipelines, covering their needs in Section 5, benefits in Section 6, challenges in Section 7, differences from unidirectional pipelines in Section 4, and the process of transitioning from unidirectional to bidirectional data pipelines in Section 8. The findings and research implications of the study are discussed and concluded in Section 9.

2. Background

In the current digital environment, the significance of data and effective data management cannot be overstated. Data has emerged as a crucial component of decision-making and strategy for organizations across various industries [16]. Data encapsulates valuable insights, patterns, and trends that, when used appropriately, help companies gain a competitive advantage [17]. Data management, in turn, serves as the basic requirement for organizing, storing, and safeguarding this asset [18]. Efficient data management ensures data accuracy, accessibility, and security, enabling a reliable basis for analytical processes [18]. As the volume, variety, and velocity of data continue to rise, the need for streamlined and efficient mechanisms to handle this data becomes critical [19] [12]. This is where data pipelines come into play, representing the arteries of a data-driven ecosystem. Data pipelines orchestrate the seamless flow of data from diverse sources to destinations, enabling organizations to extract, transform, and load data for analysis [12]. They constitute a vital infrastructure that not only improves operational efficiency but also facilitates real-time decision-making [5]. To cut deeper into the significance of data pipelines, it is essential to explore their role in managing the dynamic and complex nature of modern data ecosystems, addressing challenges, and leveraging trends to unlock the full potential of data as a strategic asset [5].

As we delve into the complexities of data management, data pipelines emerge as a crucial and significant component within data-driven environments [5]. They serve as the connective tissue between various data sources and destinations, ensuring a smooth flow of information for analysis and decision-making [12]. The significance of data pipelines lies in their ability to automate and streamline the complex process of collecting, transforming, and delivering data to its intended endpoints [5]. This automation not only improves operational efficiency but also reduces the likelihood of errors and ensures data consistency. In essence, data pipelines act as the arteries of a dynamic data ecosystem, enabling organizations to harness the full potential of their data assets by making the information readily available for analysis and insights [20]. Their role in optimizing the data workflow makes data pipelines a linchpin in the broader field of data management, fostering a more agile, responsive, and scalable approach to handling the ever-growing volume and complexity of data in today's digital landscape [21].

Several trends, advancements, and challenges in the field of data management have collectively propelled the need for extensive research on data pipelines. The exponential growth in data volume and the increasing variety of data sources, including structured and unstructured data, demand more sophisticated and scalable solutions [22]. Data pipelines provide a structured way to handle diverse data types and large datasets efficiently [5]. Organizations require immediate insights for timely decision-making. Data pipelines equipped to handle streaming data enable the processing of information as it is generated, allowing for real-time analysis [23].

With the rise of edge computing, data is being generated and processed closer to the source and requires data pipelines that can efficiently manage the bidirectional flow of data between edge devices and central systems, ensuring seamless integration and analysis [24]. The growing adoption of advanced analytics, machine learning, and artificial intelligence necessitates data pipelines that support the training and deployment of models [25]. These pipelines must accommodate the iterative nature of machine learning workflows and facilitate the integration of insights back into

operational systems. Data pipelines need to be adaptable to hybrid and multi-cloud environments, allowing organizations to leverage the flexibility and scalability offered by cloud platforms [26]. Data pipelines seamlessly integrate with various data sources, storage systems, and analytics tools to provide a cohesive and interoperable data management infrastructure [19].

In our previous study [12], we designed a conceptual model for data pipelines, which can aid in communication between different data teams and automate monitoring and fault detection. Advancements in automation and orchestration technologies are driving the need for fault-tolerant data pipelines. Further, our recent research on data pipelines has identified key challenges such as data quality issues, infrastructure maintenance problems, and organizational barriers, while also emphasizing the benefits of traceability, fault tolerance, and reduced human errors [5]. Oleghe et al. [27] developed a framework for designing data pipelines in manufacturing systems, providing a template for selecting key layers and components. Dakkak et al. [28] identified the relation between data dimensions and collection challenges, proposing an architecture containing three collection levels: local, regional, and global. Kosar et al. proposed data pipelines as an automated system for transferring large-scale multiprotocol data, demonstrating their effectiveness in transferring terabytes of data without human intervention [19]. These studies collectively underscore the importance of data pipelines in enhancing data processing and transfer efficiency.

Data pipelines can be made resilient and responsive by incorporating two-way data flow. They can also provide a foundation for handling diverse data sources, supporting real-time analytics, managing edge computing scenarios, facilitating advanced analytics, ensuring data security, promoting interoperability, and optimizing operational efficiency—all of which are critical components of an effective and adaptive data management strategy in today's dynamic business environment. However, limited academic attention has been given to this subject, as evidenced by the scarcity of relevant research papers. Therefore, we provide a comprehensive study on bidirectional data pipelines.

3. Research Methodology

This research is based on multiple case studies conducted with three data engineering teams at one company. Each team works with customers in a specific geography around the world to establish and maintain customer-specific data pipelines. Multiple-case study research methodology involves the investigation of several cases to gain a comprehensive understanding of a research question [29]. This approach allows researchers to analyze similarities and differences across multiple cases, facilitating the identification of patterns, trends, and generalizable findings [30].

To set the scope for the type of empirical studies we address in this paper, we have used a three-step research model. In the first step, we presented our previous research on the topic of data pipelines [21, 10, 5] in a company workshop where the members of the three teams were also present. After the presentation, we got to know about the bidirectional data pipelines at the company. Then we conducted a literature review on the topic to understand the existing body of knowledge and identify gaps in the literature. As the second step, we conducted semi-structured interviews with selected members of each team. The second author, an employee of the case company, helped choose the participants for the interview. The first author, being an embedded researcher in the company, got the opportunity to analyze the documents related to the topic. Finally, the authors performed analysis and synthesis to present the results.

3.1. Research Questions

The primary research objective of this study is to conduct an in-depth investigation into the necessity of bidirectional data pipelines in modern data-driven environments, assess their benefits, analyze the challenges associated with their implementation and management, and compare the key differences between unidirectional and bidirectional data pipelines. Additionally, the study aims to outline a systematic framework for transitioning from unidirectional to bidirectional data pipelines, emphasizing the crucial steps and considerations essential for a seamless and effective migration process. We developed the following research questions to achieve the objectives discussed in Section 1.

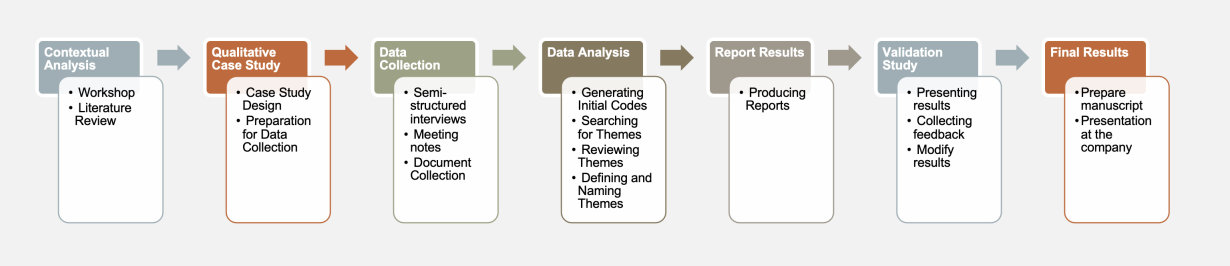

Figure 1: Research Process

RQ1. *How is a bidirectional data pipeline different from the combination of two unidirectional data pipelines?*

RQ2. *What are the benefits and challenges encountered in the implementation and management of bidirectional data pipelines?*

RQ3. *What are the critical considerations and strategies for successfully transitioning from unidirectional to bidirectional data pipelines?*

To answer these research questions, we conducted an exploratory multiple-case study with multiple practitioners in several organizations. The use cases and analysis are explained in the below subsections. Data collection and data analysis are described in Sections 2.3 and 2.4, respectively. Fig. 1 shows an overview of data collection and data analysis.

3.2. Qualitative Case Study

This research was based on a qualitative case study design for the following reasons: The first reason is that qualitative case studies allow researchers to gain a comprehensive and detailed understanding of the intricate dynamics and complexities of bidirectional data pipelines in industrial settings [30]. This approach facilitates the exploration of the various contextual factors and challenges that may arise during the implementation and utilization of these pipelines. Additionally, given that the concept of bidirectional data pipelines has not been thoroughly explored in existing research, utilizing a case study method is deemed suitable for gaining insights into this specific phenomenon in an industrial setting. Second, case studies enable researchers to examine bidirectional data pipelines within their specific industrial contexts, taking into account the unique organizational structures, processes, and constraints that may influence the design, implementation, and performance of these pipelines [31]. Third, the qualitative approach is effective in exploring the multifaceted nature of bidirectional data pipelines, including the diverse technical, operational, and organizational aspects that contribute to their functioning and effectiveness within industrial environments. Fourth, qualitative case studies facilitate the collection of data from various stakeholders involved in the development, deployment, and management of bidirectional data pipelines. This allows researchers to capture different perspectives and insights from practitioners, engineers, and decision-makers, providing a holistic view of the challenges and opportunities associated with these pipelines [30]. Fifth, bidirectional data pipelines often operate in dynamic and evolving industrial settings. Qualitative case studies offer the flexibility to adapt research methods and data collection techniques to capture changes and developments in real time, enabling researchers to assess the adaptability and resilience of these pipelines [32]. Finally, through qualitative case studies, researchers can identify challenges and benefits related to the implementation and maintenance of bidirectional data pipelines in industrial contexts, offering valuable insights for industry professionals and stakeholders.

To ensure a systematic approach, we adhered to the five steps outlined for a software engineering case study, following the guidelines proposed by Runeson et al. [33]

1. **Case Study Design:** This step involves defining the objectives of the case study and planning the overall approach. Researchers outline the scope of the study, specify the research questions, and establish the criteria for selecting the cases to be included in the study.

2. **Preparation for Data Collection:** In this step, researchers define the procedures and protocols for data collection. This includes identifying the sources of data, determining the data collection methods (such as interviews, observations, or document analysis), and creating a framework for systematically gathering relevant information.
3. **Collecting Evidence:** The third step entails the execution of the data collection procedures defined in the previous step. Researchers collect empirical evidence from the selected cases, ensuring that the data collected is aligned with the research objectives and addresses the identified research questions.
4. **Analysis of the Collected Data:** Once the data has been collected, researchers analyze the gathered evidence using various qualitative or quantitative analysis techniques. This step involves examining the data for patterns, themes, and insights that can answer the research questions and contribute to a deeper understanding of the phenomenon under investigation.
5. **Reporting of the Results:** The final step involves presenting the findings and results derived from the data analysis. Researchers document the outcomes of the case study, including the key insights, implications, and recommendations. Clear and comprehensive reporting is essential for communicating the research outcomes effectively to both academic and industrial audiences.

3.3. The case company

The choice of the company as a case company is primarily due to its prominence in the telecommunications and information technology industries, as well as its extensive experience in developing and deploying complex data infrastructure and networking solutions.

The company is a leading global provider of telecommunications equipment and services, with a strong reputation for innovation and technological expertise in the field of communication networks. Its involvement in developing advanced data pipelines and networking solutions makes it a compelling subject for research on bidirectional data pipelines. The company's significant technological capabilities and research and development initiatives in areas such as 5G, cloud computing, and the Internet of Things (IoT) offer valuable insights into the design, implementation, and management of bidirectional data pipelines within complex and dynamic environments. Further, the company serves a diverse client base, comprising telecommunications operators, enterprises, and governments globally. Research conducted on the company's data pipelines can provide insights into the challenges and requirements of managing bidirectional data flows in various operational contexts and industries. Furthermore, with its global reach and widespread influence in the telecommunications industry, insights derived from studying the company's approach to bidirectional data pipelines can have far-reaching implications for the development of industry standards and innovative solutions in the fields of data management and communication networks.

Although this case study was conducted with a single company, we investigated bidirectional data pipelines with multiple teams spread over six locations in four countries.

3.4. Use cases

We analyzed the characteristics of data flowing through the pipeline in both directions and found that the data pipelines can be represented as a staircase model, as shown in Fig. 2. We made the model such that it illustrates the distinct stages in the development and transition of a data pipeline from functioning in a unidirectional manner to operating in a bidirectional fashion, and mapped our use cases to corresponding stages as shown in Fig. 2. These use cases are instrumental in demonstrating the practical implementation of data pipelines in the context of customer service operations. Moreover, as shown in Table 1, they collectively provide a comprehensive overview of the evolutionary journey of data pipelines. By presenting diverse use cases that embody the transformation from unidirectional to bidirectional data flow, the study aims to illustrate the developmental trajectory and the varying degrees of interactivity and feedback integration that occur within data pipelines over time. In the first stage, data flows only in one direction. The second stage, which is Limited Bidirectionality has data flowing from source to destination, and the destination sends analytic feedback to the source. The third stage, which is Balanced Bidirectionality has data flowing from source to destination, and the

destination sends anomaly alerts to the source. The fourth stage, which is Enhanced Bidirectionality has data flowing from source to destination, and the destination is capable of executing commands at the source. Finally, in full bidirectional integration stage, data flow happens in both directions.

Use case 1: Adaptive Data Collection and Targeted Analysis for Efficient Problem Resolution - The company optimizes data collection and analysis by implementing an adaptive approach, collecting a baseline amount of data, and making inferences to identify potential issues. Based on these inferences, it selectively collects additional information on a targeted scale for focused interrogation and detailed analysis. Fig 3 shows the key components of the use case.

Team D collects an initial baseline amount of data from nodes, considering network constraints, hardware limitations, and cost considerations. Essential inferences are made based on the initial data, identifying potential issues or anomalies that may require further investigation. Also, triggers are set based on inferred issues to selectively initiate additional data collection on a targeted scale, avoiding the need to transfer large volumes of data constantly. If required, cast a wide net and collect a broader set of information when triggered, capturing a wide swath of data to explore potential issues in more detail. If anything interesting is identified within the wide net data, perform a very focused additional interrogation of specific nodes, parts of nodes, services, or subsystems. Then, a detailed and focused analysis is conducted on the selectively collected data, gaining a more comprehensive understanding of the identified issues. Based on the detailed analysis, recommendations are generated for problem resolution, optimization, or further investigation. Data transfer is optimized by avoiding constant large-scale data transfer, ensuring cost-effectiveness and efficient use of network and hardware resources.

Direct quotes: *"Our team collects data and then makes essential inferences on that information, which gets stored in the nodes. However, in many cases, the amount of data produced is not practical to constantly transfer due to network constraints, hardware constraints, and cost constraints. Further, our team doesn't have the ability to necessarily exfiltrate all the information. Therefore, we essentially collect some baseline amount of data, and based on the baseline, we make inferences to say if there are any problems that need them to collect additional information on a targeted scale. Then we collect a wide swath of information, cast a wide net, and if we see anything interesting within that wide net, do a very focused additional interrogation of data of particular nodes or particular parts of nodes or services or subsystems to get a more detailed view because it wouldn't be practical to collect the data from all those subsystems essentially all the time"* - P1

Use case 2: Enhanced Network Performance Monitoring with Anomaly Detection - A robust system is implemented for data collection, processing, and analysis to monitor and optimize network performance utilizing various anomaly detection techniques, including median absolute deviation and DB scan, to identify abnormal behavior and potential network change events. Fig. 4 shows the key components of the use case.

Team A initiates' data collection transfers over a secure tunnel between the company and the customer, ensuring the seamless and reliable transfer of data. Container services are utilized running on the company server for processing and loading collected data into a dedicated database. Team A also maintains a database for storing counters and relevant performance data, facilitating direct data interaction and use case creation within the ENI system. Counters are loaded into the database and generate basic Key Performance Indicator (KPI) reports for individual nodes or clusters, providing insights into network performance. Worst offender analysis is conducted on nodes or clusters, identifying and addressing performance issues that have the most significant impact. Further, various anomaly detection techniques, including median absolute deviation and DB scan, are implemented to identify abnormal behavior in network performance data. Current performance data is compared with historical performance, considering seasonality and weekday versus weekend differences. Several weeks of data are used to establish a baseline for network performance, taking into account normal variations and trends. Network performance is continuously monitored, and change events are detected by assessing deviations from the established baseline. Finally, detected abnormal behavior is analyzed to understand potential causes and implications for network performance.

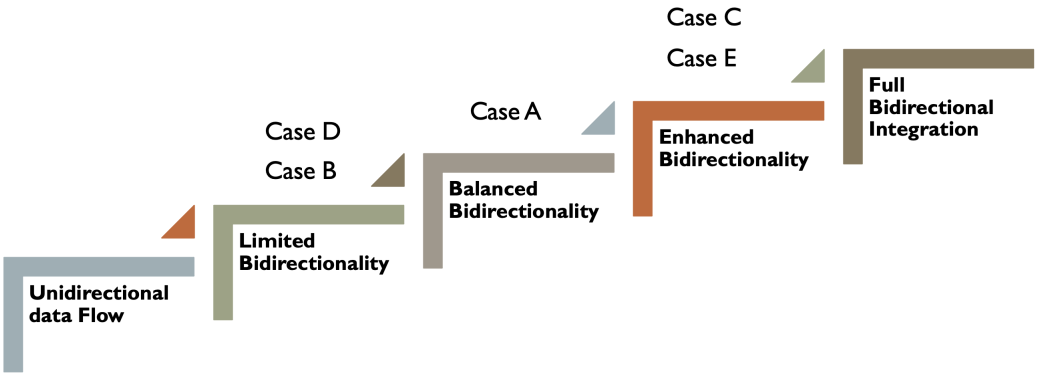

Figure 2: Use case mapping to the transition stages

Table 1: Use cases and description

Use case ID	Use case	Description	Category	Interviewed Experts			
				ID	Role	Years of Experience	Team
1	Adaptive Data Collection and Targeted Analysis for Efficient Problem Resolution	Data collection and analysis by implementing an adaptive approach, collecting a baseline amount of data, and making inferences to identify potential issues	Limited Bidirectionality	P1	System Solution Architect	over 10 years	D
2	Enhanced Network Performance Monitoring with Anomaly Detection	Data is collected, processed, and analyzed to monitor and optimize network performance utilizing various anomaly detection techniques	Balanced Bidirectionality	P2	System Solution Architect	over 15 years	A
3	On-Demand Performance Analysis with Data Collection Server Integration	Collects and analyzes performance management data, configuration management data, and event management data from customer premises and implements an on-demand structure to address real-time analysis requirements	Limited Bidirectionality	P3	System Solution Architect	over 15 years	B
4	Enhanced Data Collection Resilience and Integrity for Network Operations	Data pipeline to ensure the continuous availability, integrity, and resilience of data collection processes, particularly in the networking domain	Enhanced Bidirectionality	P4	System Solution Architect	over 5 years	C
5	Flexible Data Pipeline for Network-Related Data Processing and Analysis	A versatile data pipeline capable of ingesting various types of network-related data and storing it in a centralized data store or data lake for seamless data interaction	Enhanced Bidirectionality	P5	System Solution Architect	over 5 years	E

Direct quotes: *"We have our system that does its data collection transfers over a tunnel between the company and the customer, and then we have different container services running on a server in the company that process and load this data into a database. I generally work directly with the data in the database or create our use cases within the ENI system. We're loading counters into this database that we maintain, and we either generate just basic KPI reports for nodes or clusters like worst-offender type analysis. We run use cases that use that data and do different types of anomaly detection, such as median absolute deviation and DB scans, trying to compare performance to historical performance. We try to consider seasonality and weekday versus weekend differences. It's usually using several weeks of data and trying to essentially establish a baseline and determine if the network changes if that baseline changes and there's some abnormal behavior."* - P2

Use case 3: On-Demand Performance Analysis with Data Collection Server Integration The team efficiently collects and analyzes performance management data, configuration management data, and event management data from customer premises and implements an on-demand structure to address real-time analysis requirements, establishing seamless communication between the Data Collection Server (DCS), worker module, and nodes. Fig 5 shows the key components of the use case.

First, team B collects performance management, configuration management, and event management data from customer premises at different frequencies. Pricing and performance management data are analyzed to identify dips or peaks that require further investigation. An on-

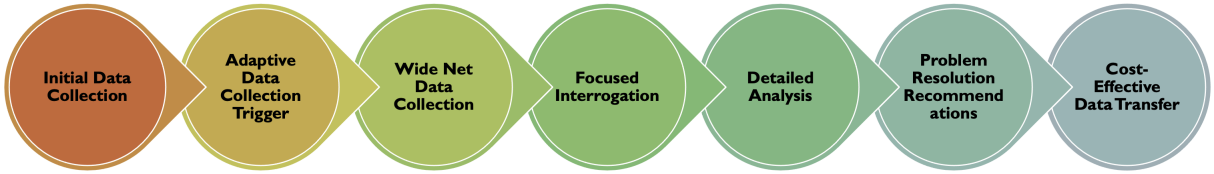

Figure 3: Use case 1 - Adaptive Data Collection and Targeted Analysis for Efficient Problem Resolution

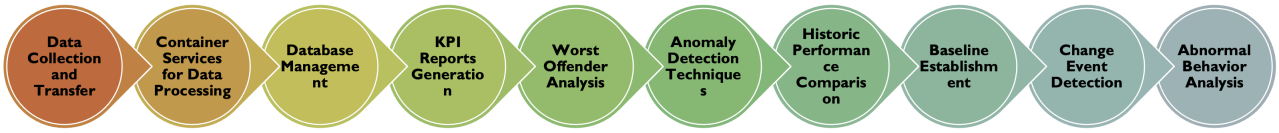

Figure 4: Use case 2 - Enhanced Network Performance Monitoring with Anomaly Detection

demand structure is implemented for real-time analysis, allowing dynamic requests for specific node information. Communication channels are established between the Data Collection Server (DCS), worker module, and nodes. When an on-demand request is initiated, the worker server puts the request in a message queue. Then the request is transferred to the DCS, which collects data in batches, executing simultaneous requests for multiple nodes efficiently. Collected data is compressed in batch format, ensuring efficient storage and transmission. Based on user preferences, flexibility in data format is provided, supporting CSV, HTML, or raw data options. Compressed data is parsed upon response, extracting relevant information for analysis. Further, the data is structured for storage in a refined data storage system, ensuring accessibility and ease of analysis. Users can set conditions such as higher or lower thresholds for analysis results. Finally, analysis findings are generated and stored in a database, organized by date and time for future reference.

Direct quotes: *"We collect performance management data, configuration management data, and event management data from the customer premises, each at different frequencies. Sometimes when we need to analyze pricing and some of the performance management data, we are required to go to the nodes to get some real-time information. If we find that at some part of that time in the day, there is a dip or a peak in performance management data, in the next step, we need more information to see what is happening. So in that case we have an on-demand structure. So for that, we have to establish communication between we DCS (Data Collection Server) and the worker module. DCS is the server that resides at the customer location, where we have direct communication with the worker and the nodes. Let's say we require the information for the 15 nodes or the 200 nodes. Sometimes it uses batch processing so that the 20 nodes or 25 nodes will be executed once, then it plugs the data and compresses it in any format. When the response is there, it gives the message and then the worker will be able to get the data in the compressed format. We parse the data and make it structured for storing in the refined data storage for analysis. The final findings will be based on the conditions the user is setting up. So you might set a higher threshold or lower threshold depending on the customer, and those results are generated and stored in the DB so that in the future we can check it based on the date and time"* - P3

Use case 4: Enhanced Data Collection Resilience and Integrity for Network Operations The data pipeline implemented by Team C ensures the continuous availability, integrity, and resilience of data collection processes, particularly in the networking domain, to guarantee the uninterrupted functionality of upper-layer analyses and maintain the accuracy of dependent analyses. Fig 6 shows the key components of the use case.

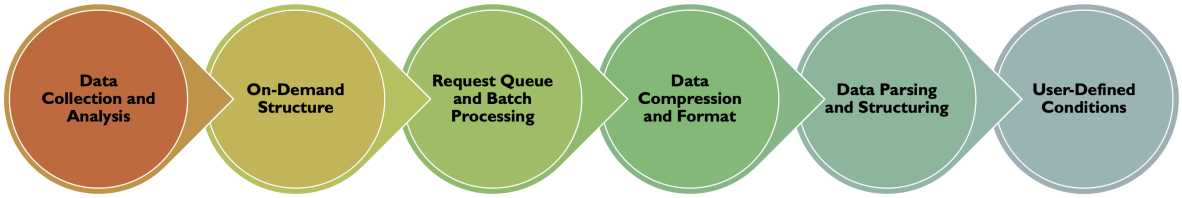

Figure 5: Use case 3 - On-Demand Performance Analysis with Data Collection Server Integration

Figure 6: Use case 4 - Enhanced Data Collection Resilience and Integrity for Network Operations

Redundancy controls and monitoring mechanisms are implemented to ensure the continuous availability of data collection processes. Resiliency features are incorporated to handle failures in data collection and enable efficient recovery. For backtracking, a robust backtracking mechanism is developed to track and recover data in the event of a collection failure. Job queuing mechanisms are integrated to prioritize and manage data collection tasks, ensuring efficient execution after recovery. To incorporate integrity checks and job queuing, an integrity check mechanism is implemented to assess the data's quality before upper-layer analysis, preventing the propagation of inaccurate or missing data. Job queuing is incorporated to manage the re-collection of data after recovery, confirming the necessity of collecting data generated an hour ago. Two-way pipeline logic is incorporated to determine the relevance of actions after a recovery event so that the use case's requirements are considered to decide whether specific actions, such as restarting a process, are still applicable after a failure and recovery cycle. An audit trail is also established to demonstrate the logic correctness and evidence of the proposed actions to customers before execution.

Direct quotes: *"I think the data collection availability is quite important, especially on the networking part. Because everything else is built on top of it. Make sure a networking part on the lower layer needs to work, either through some redundancy controls or monitoring, depending on how much it means for the business. I mean, it is important to have a resiliency part here. When data collection fails, and when it recovers and tries to backtrack the data we need to collect, it's extremely important for the upper layer analysis, where it needs a certain level of integrity of the data. Otherwise, dependent analysis doesn't work, and you will end up with a lot of missing data which affects the feedback accuracy as well. Another concern is that when collecting data after the recovery, there should be a mechanism that confirms whether the data that was generated an hour ago still needs to be collected again or not. To keep integrity and resiliency, if you apply this logic to the two-way pipelines, you may not need to execute an action. Because if I say I fail at this hour and I try to apply this action an hour later after backtracking, maybe this doesn't really apply anymore, right? Let's say. I wanted to restart the radio at this hour, but you know an hour later, but somehow something went wrong. I couldn't restart it. So an hour later, should I restart it again? Well, I think that depends on the use case. A part of the logic has to be controlled by the use case to decide.*

While implementing backtracking, just before you apply the action, typically the best practice that we use in our system is that instead of actually applying the action we sort of store that action in some database or some data storage so that we have another building block to read from it. So when this type of situation comes, we check at what time it needs to be executed and how many times you need to execute the actions" - P4

Use case 5: Flexible Data Pipeline for Network-Related Data Processing and Analysis Team E has implemented a versatile data pipeline capable of ingesting various types of network-related data and storing it in a centralized data store or data lake that can enable consumers, including Business Intelligence (BI) tools, to efficiently analyze and process the data, creating a bidirectional flow for seamless data interactions. Fig 7 shows the key components of the use case.

The data pipeline ingests different types of network-related data, including configuration management, alarm-related information, and performance management data in XML format. Then it performs data transformations to streamline diverse data types into a standardized structure, such as a comma-separated value (CSV) format, suitable for storage and analysis. The transformed data is stored in a centralized data store or data lake, supporting various storage options like a database management system, NoSQL database, and flat files. BI tools consume data from this centralized storage, analyze it, and generate meaningful insights. The results are stored back in the same data storage, creating a two-way data flow. It is a bidirectional data pipeline because data is not only collected and stored but also analyzed, and processed, and results are fed back into the central data store for further use. The data pipeline also has implemented mechanisms for real-time data streaming, allowing on-demand execution of the data pipeline to cater to scenarios where additional information is needed in real-time. Dynamic access to the data lake is supported, allowing the data pipeline to retrieve stored information based on specific use cases, enhancing flexibility and adaptability. The data pipeline is designed for scalability to handle increasing data volumes and extensibility to incorporate new types of network-related data seamlessly. Further, security measures are implemented to safeguard sensitive network data during ingestion, transformation, storage, and analysis. Ensure compliance with data privacy regulations.

Direct quotes: *"In general, our data pipeline, has the capability of absorbing different types of data. I mean exporting different types of data and then doing some kind of transformation and then loading it into a central data store or data Lake. It is a part of our ENI application created for any kind of consumer who wants to consume that centralized data. It mostly relates to different types of network-related data. In most cases, it does not, basically, look for data in any kind of database or data storage. So it's mostly about the collection of data from different, networks and then getting those data either through a data streaming mechanism or by pulling data from a specified path or location, and then the transformation happens, and the data gets stored inside a mini IO data link. Let's say we have a database management system, we have a no-SQL database, and we have some flat files we collect. They streamline to a simple data store, a data structure, and put it into data storage. We collect configuration management data. We collect alarm-related information. We collect performance management data, which is in the form of XML files. Then we basically streamline those data and create a comma-separated value structure for now, and then it gets stored inside the data link. One of the customers for this data is a BI tool that takes data from the storage, analyses and processes the data, and stores the results back in the same storage, which is one aspect of our two-way data pipeline. In some other use cases, the starting point of the use case is already stored in the data lake. Based on that data, it may again need some additional information that may not be available within our existing data link. So we need to do an on-demand or a real-time execution of the data pipeline streaming again."* - P5

3.5. Data Collection

The data compiled for this study encompassed various sources, including transcripts of semi-structured interviews, meeting notes, emails, documentation, and presentations. The primary researcher worked closely with the company employees, being an embedded researcher in the company. Meanwhile, the second author, an employee of the company, played a significant role in selecting interview participants and actively participated in the research and interview meetings.

Figure 7: Use case 5 - Flexible Data Pipeline for Network-Related Data Processing and Analysis

To ensure a comprehensive sampling approach, we combined criterion sampling with convenience sampling, selecting knowledgeable practitioners willing to participate in the study. The interviewees were drawn from different roles, with varying levels of experience ranging from 3 to 25 years, across six locations in four countries. The semi-structured individual interviews were conducted through phone conferences, with interview durations ranging from 50 minutes to 1 hour and 30 minutes. These interviews took place between Feb 2022 and July 2022. Both authors were present during all interviews, with tailored interview guides designed to extract relevant insights based on the participants' roles and expertise.

The interviews covered a range of topics, including the interviewees' backgrounds, roles within the organization, and their experience with bidirectional data pipelines or similar data management systems. Additionally, we delved into the procedures and methodologies used for data validation and quality assurance within the bidirectional data pipelines, data verification processes, error handling mechanisms, and data consistency checks. The discussions also touched on challenges and specific requirements that practitioners encounter when designing and implementing bidirectional data pipelines in their respective industries. Further, we explored the best practices, lessons learned, and practical recommendations for effectively managing and optimizing bidirectional data pipelines. Finally, we also encouraged practitioners to share their experiences and success stories.

3.6. Data Analysis

The collected data underwent thematic coding using the qualitative analysis software NVivo, following the six-phase process proposed by Braun and Clark [34] for thematic analysis. Inductive thematic coding was employed, allowing themes to be derived directly from the data rather than being guided by preconceived coding frames.

The initial phase involved the researchers becoming acquainted with the data through active participation in the interview and transcription processes, as well as through thorough readings of the transcriptions and interview notes. The first author thoroughly read the data to become familiar with its intricacies and identify potential themes.

In the subsequent phases, the first author generated a set of codes such as real-time responsiveness, closed-loop, security, consistency, scalability, conflicts, flexibility, decision-making, latency, customer experience, automation, two-way data flow, advantage, challenge, etc., representing significant concepts and ideas identified during the interviews and discussions in additional documents. Then, the codes were refined by the first author by modifying and/or renaming some of them. Further, the first author and second author examined the refined codes to identify overarching themes that encapsulate patterns and concepts within the data. These codes encompassed various aspects, such as the research objectives, the importance of bidirectional data pipelines, differences between unidirectional and bidirectional data pipelines, perceived benefits and challenges, and transition milestones.

Themes were then identified through collaborative discussions in the third phase, followed by a review and consolidation of potential themes and related codes in the fourth phase. The interviewees, their roles, locations, and years of experience are summarized in Table 1, showcasing the diverse perspectives gathered during the study.

In the fifth phase, each theme was meticulously analyzed to extract the main findings and discussion points relevant to addressing the research questions. Additionally, the authors carefully selected, summarized, and anonymized company-specific information to provide clear examples for the research paper, ensuring the confidentiality of

the company data.

In the sixth phase, the authors conducted a second round of discussion with selected participants in December 2023 to validate the results of the study. The selection was based on years of experience working with data pipelines. The second author, who is also an employee at the case company, suggested the participants and arranged interviews with them. The first author presented the results to the interviewees, collected feedback, and modified the manuscript, which was revised by the third and fourth authors. Further, the second author submitted the paper for approval from the company, and the results were also presented by the authors at the company.

The final phase involved the publication of the research findings and the explicit addressing of the research questions in the discussion section.

4. Comparison between unidirectional data pipelines without shared data transmission channel and bidirectional data pipelines (RQ1)

Unidirectional and bidirectional data pipelines represent two different approaches to data transfer between systems or applications. Based on the empirical study, understanding the differences between unidirectional and bidirectional data pipelines is essential for determining the most suitable approach based on the specific data management requirements and use cases of an organization. Each type of pipeline has its advantages and is appropriate for different data transfer and synchronization needs. Through our study, we found that there could be situations where the pipelines cross the same transit nodes or share the flow in some segments of the pipeline, but the essence here is that they are logically separated even if the physical medium is the same.

4.1. Two Unidirectional Data Pipelines without shared data transmission channel

As the name indicates, the two-way data flow is enabled by connecting two separate unidirectional data pipelines in opposite directions. Each pipeline operates independently, maintaining its unidirectional nature, but the connection enables data to move in opposite directions. This setup provides flexibility, allowing different types of data or tasks to flow in distinct directions while still having some level of bidirectional communication. It's a useful architecture when specific functions or tasks require separate data flows but benefit from the bidirectional connection for coordination.

A primary pipeline, a secondary pipeline, and a connector or middleware are the general components of this type of data pipeline. The primary pipeline is the first unidirectional data pipeline and often includes all the components of a standard unidirectional pipeline: source, ingestion, processing, storage, destination, monitoring, and logging. The second unidirectional data pipeline, known as the "secondary pipeline," operates independently but in the opposite direction. From the use cases discussed in Section 3 the secondary pipelines usually deal with analytic reports, anomaly alerts, or commands that should be executed at the other end. A middleware or connector facilitates the exchange of data between the primary and secondary pipelines. It acts as a bridge, translating and transmitting data between the two systems.

The data originates from the source in the primary pipeline. The data is ingested, processed, and stored in the primary pipeline. The connector, or middleware, extracts relevant data from the primary pipeline and transmits it to the secondary pipeline. The data originates from the source in the secondary pipeline. The connector, or middleware, extracts relevant data from the secondary pipeline and transmits it to the primary pipeline. Two-way data pipelines often require synchronization mechanisms to ensure that both systems have consistent and up-to-date data. Synchronization may involve handling conflicts, managing updates, and maintaining data integrity across both pipelines.

In summary, connecting two unidirectional data pipelines in opposite directions introduces a dynamic and responsive data flow that enables real-time collaboration, data replication, and seamless integration across diverse systems. While this bidirectional setup offers increased flexibility, it also comes with challenges related to conflict resolution, latency, and data consistency. Organizations adopting bidirectional data pipelines must carefully design and implement solutions to address these challenges and ensure the reliability and integrity of their interconnected data ecosystem. All data pipelines except use cases 3 and 5 discussed in Section 3 belong to this category.

4.2. Bidirectional Data Pipelines

Bidirectional data pipelines support data flow in both directions, allowing for interaction and communication between the source and destination. Unlike unidirectional pipelines, bidirectional pipelines facilitate feedback mechanisms, enabling real-time synchronization, mutual updates, and continuous interaction. These pipelines are crucial in scenarios where dynamic, two-way communication is essential, such as collaborative applications, real-time systems, or any environment requiring constant updates and responses. The bidirectional nature enhances the responsiveness and adaptability of the data flow.

Components of Bidirectional Data Pipelines are Source Systems, Ingestion, Processing, Storage, Destination Systems, Connector or Middleware, Monitoring and Logging. Each end of the bidirectional pipeline has its source system. These sources could be databases, applications, IoT devices, or any data-producing entities. Ingestion processes on both ends of the pipeline are responsible for collecting and importing data from the source systems. This may involve ETL (Extract, Transform, Load) processes to prepare the data for further processing. Both pipelines include processing components to transform, enrich, or analyze the incoming data. Business rules, data validations, or other transformations may be applied at this stage. Processed data is stored at each end, ensuring that both systems have access to relevant and up-to-date information. Storage may involve databases, data warehouses, or other suitable repositories. On both ends, there are destination systems where the processed data is utilized. These could be applications, analytics platforms, reporting tools, or any other systems that consume the data for decision-making. The bidirectional flow is facilitated by a connector or middleware that acts as a bridge between the two pipelines. This component is responsible for translating and transmitting data bidirectionally, ensuring consistency and synchronization. Both pipelines require monitoring and logging components to track the health, performance, and any issues that may arise during bidirectional data exchange.

Bidirectional pipelines enable dynamic interactions between systems, supporting real-time collaboration and synchronization. Maintaining data consistency between the two systems is a key focus. Synchronization mechanisms ensure that both ends have the latest version of the data. Bidirectional pipelines offer flexibility in how data is exchanged and updated. Systems can both provide and consume information as needed. In scenarios where real-time updates are crucial, bidirectional pipelines support the continuous exchange of data between systems.

4.3. General Challenges

Conflict Resolution: Handling conflicts that arise when updates occur simultaneously in both pipelines is a critical challenge. Implementing effective conflict resolution mechanisms is essential to maintain data integrity.

"We have incorporated integrity checks and job queuing, to assess the data's quality before upper-layer analysis, preventing the propagation of inaccurate or missing data" -P4

Latency and Performance: Data pipelines with two-way data flow may introduce additional latency due to the need for synchronization. Ensuring optimal performance and responsiveness requires careful design and optimization.

"While implementing backtracking, typically the best practice that we use in our system is that instead of actually applying the action, we sort of store that action in some database or some data storage so that we have another building block to read from it. So when this type of situation comes, we check at what time it needs to be executed and how many times you need to execute actions" - P4

Data Consistency: Ensuring consistent and accurate data between the two pipelines is crucial. Inconsistent data can lead to errors and discrepancies in downstream applications.

"Data collected after failure recovery might have lost its context and relevance. At a particular timeframe, we will be expecting certain data and together with that if old data is received, it will eventually lead to data inconsistency" - P1

Security Concerns: Data pipelines with two-way data flow require robust security measures to protect the transmitted data. Encryption, authentication, and access controls are critical components of a secure data pipeline with two-way data flow.

"I feel that security is super important in fortifying the integrity and confidentiality of the data being transmitted through the pipeline. So, we have authentication mechanisms implemented in at least one direction in our data pipelines"

In short, bidirectional data pipelines provide a powerful solution for scenarios where data needs to flow dynamically between systems in both directions. They support real-time collaboration, master data management, and integration in complex, distributed environments. However, organizations implementing bidirectional pipelines must carefully address challenges related to conflict resolution, latency, data consistency, and security to ensure the reliability and effectiveness of the bidirectional data exchange. Use cases 3 and 5 described in Section 3 are examples of this type of data pipeline.

Table 2 shows the key distinctions between these two types of pipelines:

5. Significance of Bidirectional Data Pipelines (RQ2)

Bidirectional data pipelines have become increasingly essential for organizations. From the empirical study based on the use cases discussed in section 3, we have identified the following key reasons:

5.1. Enhanced Data Consistency

Bidirectional pipelines help maintain consistency across different systems by ensuring that any changes or updates made in one system are automatically reflected in another. This significantly reduces the chances of data inconsistencies and discrepancies, which can lead to errors and confusion.

"Bidirectional data pipelines are essential for maintaining enhanced data consistency, ensuring that accurate and up-to-date information is available across various systems and applications. This capability is vital for us, who rely on accurate and consistent data to make informed business decisions and drive organizational success." - P3

5.2. Improved Workflow Efficiency

By enabling the continuous flow of data in both directions, bidirectional data pipelines streamline workflow processes. This leads to increased operational efficiency and productivity, as employees can access and utilize the most updated information without delays or manual interventions.

"Ideally, these two-way pipelines should enable real-time data updates, automated data transfer, streamlined collaboration, faster response to changes, integration of disparate systems, and optimized resource allocation, ultimately contributing to a more efficient and productive working environment. - P1

5.3. Faster Decision-Making:

Access to real-time, synchronized data empowers organizations to make timely and informed decisions. Bidirectional data pipelines enable stakeholders to access the latest data from various sources, facilitating quicker and more accurate decision-making processes.

"Bidirectional data pipelines play a significant role in accelerating the data processing and integration cycle, helping us to access real-time, accurate information for making timely and informed decisions." - P3

5.4. Facilitation of Data-Driven Strategies:

In the era of data-driven decision-making, bidirectional data pipelines play a crucial role in enabling organizations to leverage data effectively. They facilitate the seamless flow of information, which is essential for implementing data-driven strategies and deriving actionable insights.

"Enable us to harness the power of data effectively, empowering us to make informed decisions and formulate data-driven strategies." - P2

5.5. Real-Time Data Synchronization

In many contexts, real-time synchronization is critical to ensure that the most up-to-date information is available for decision-making. Bidirectional data pipelines enable the seamless transfer of data in both directions, ensuring that changes made in one system are reflected in the other in real-time.

"These pipelines help us to ensure that data across different systems and applications is continuously updated and consistent, enabling practitioners to access the most recent and accurate information." - P1

5.6. Seamless Integration of Disparate Systems

Many organizations use a variety of software systems and applications that need to communicate with each other. Bidirectional data pipelines facilitate the seamless integration of disparate systems, allowing data to be shared and utilized

Table 2: Comparison between unidirectional data pipelines and bidirectional data pipelines

Aspect	**Two Unidirectional Data Pipelines Connected in Opposite Directions**	**Bidirectional data pipelines**
Nature of Data Flow	Unidirectional - Data flows independently in each pipeline.	Bidirectional - Data flows in both directions between systems.
Connection Direction	Connected in opposite directions, suggesting a link between two independent pipelines.	Connected bidirectionally, indicating dynamic interaction between the pipelines.
Isolation	Relatively isolated - Changes in one pipeline may not directly affect the other.	More interconnected - Systems can send and receive updates bidirectionally.
Synchronization	Typically, less emphasis on synchronization between the two pipelines.	Emphasizes synchronization mechanisms to ensure consistency in bidirectional data flow.
Flexibility	Limited flexibility as data flows independently in each pipeline.	Offers flexibility for systems to both provide and consume information bidirectionally.
Use Cases	1. Adaptive Data Collection and Targeted Analysis for Efficient Problem Resolution 2. Enhanced Network Performance Monitoring with Anomaly Detection 3. Enhanced Data Collection Resilience and Integrity for Network Operations	1. On-Demand Performance Analysis with Data Collection Server Integration 2. Flexible Data Pipeline for Network-Related Data Processing and Analysis
Connector	May not require a specific connector or middleware to facilitate communication.	Often involves a connector or middleware to enable bidirectional data exchange.
Challenges	Coordination between independent data flows.	Managing synchronization and potential conflicts.
Dependency	Each pipeline can operate independently without direct dependency on the other.	Bidirectional data flows may have dependencies requiring careful management.

across different platforms, applications, and databases.

"Bidirectional pipelines facilitate the exchange of information between different systems and applications, enabling the seamless flow of data across various organizational processes." - P2

5.7. Enhanced Customer Experience

Bidirectional data pipelines can contribute to an improved customer experience by ensuring that customer data is up-to-date across all platforms. This leads to more personalized and relevant interactions with customers, enhancing satisfaction and loyalty.

"We need two-way data flow to deliver personalized interactions, ensure a consistent customer journey, facilitate timely issue resolution, provide proactive customer support, enable efficient communication, and utilize customer feedback effectively, ultimately contributing to an improved overall customer experience and increased customer satisfaction."-P3

5.8. Streamlined Data Management

Bidirectional data pipelines automate the transfer of data between systems, reducing the need for manual data entry and minimizing the risk of human errors. This automation increases data accuracy and reliability while freeing up resources to focus on more value-added tasks.

"We are seeking to optimize data processing, ensure data consistency and accuracy, enhance data security, and adhere to robust data governance practices." - P5

By addressing these needs, bidirectional data pipelines have become indispensable tools for organizations looking to streamline operations, improve data management, and stay competitive in today's fast-paced and data-centric business environment.

6. Benefits of Bidirectional Data Pipelines (RQ2)

Based on our empirical study, we have identified the following benefits offered by bidirectional data pipelines that make them valuable in various data management and operational contexts:

6.1. Real-time Synchronization

Real-time data synchronization between interconnected systems is made possible by bidirectional data pipelines, which guarantee that modifications made in one system instantly affect the other and vice versa. This real-time synchronization contributes to the accuracy and consistency of data across various platforms.

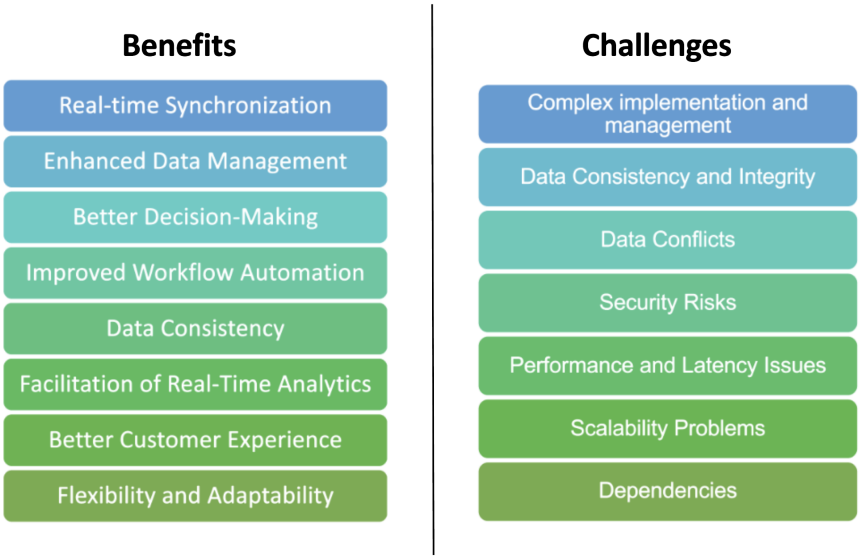

Figure 8: Benefits and Challenges of bidirectional data pipelines

"Real-time synchronization aligns seamlessly with our data-driven architectures, offering the flexibility to respond dynamically to changes in the data, which not only improves system efficiency but also supports more reactive and adaptive system behavior. If I think from a strategic perspective, real-time synchronization enhances collaboration and facilitates better decision-making processes. Teams can confidently rely on the data, promoting a collaborative environment where different teams can work cohesively toward shared objectives." - P3

6.2. Enhanced Data Management

Bidirectional data pipelines optimize data management processes by allowing data to flow seamlessly in both directions. They improve overall operational efficiency by streamlining information transfer and reducing delays and data inconsistencies.

"By incorporating data validation, cleansing, and fault detection processes into bidirectional pipelines, organizations can ensure that only high-quality data is propagated across systems. This proactive approach minimizes errors, improves the accuracy of analytics and reporting, and enhances overall data quality." - P1

6.3. Better Decision-Making

Users are better equipped to act quickly and intelligently when they have access to synchronized and up-to-date information. Decision-makers can promptly achieve more informed and precise decisions by using bidirectional data pipelines, which give them access to the most recent data from multiple sources.

"A lot of decision-making comes from receiving the data in the first place. The bidirectional part is then how you would then act on that decision, or automate acting on that decision. I mean, I think this still applies in general, that you get better decision-making from this bidirectional pipeline." - P2

6.4. Improved Workflow Automation

Bidirectional data pipelines minimize error risk and manual intervention by automating data updates and transmission between systems. Workflow procedures are streamlined by this automation, allowing employees to concentrate on more value-added tasks.

"Basically, when you automate the transmission between two systems, the workflow will be improved. Yeah, your workflow now can round trip, collect data, make decisions, and take action." - P3

6.5. Improve Data Consistency

The consistency of data across interconnected systems is guaranteed by bidirectional data pipelines. By ensuring that all linked systems

have access to the most recent information and assisting in the prevention of data discrepancies, they enhance operational consistency and lower the risk of errors.

"I mean, it depends a lot on what the system is, right, or what the objective is like, this could help you, like having the spy directional pipeline could help you improve that consistency. Like what you're doing, if you're pulling data from somewhere, that data may not be consistent. You may not have the information to determine if there's a consistency problem to push anything back, but if that's the purpose of your system, it's improving some data consistency somewhere. Then this would facilitate it. So I would say this would be on a case-by-case basis. You're going to have a system that uses the bidirectional pipeline to ensure consistency between two items and then downstream from that, you may have some other bidirectional pipeline that relies on the consistency of the source to then provide insights and actions back to that source." - P2

6.6. Facilitation of Real-Time Analytics

Bidirectional data pipelines facilitate real-time analytics by making it possible for data to be transferred between systems on time. In data-driven industries, where prompt access to accurate data is essential for making prudent business decisions, this capability is especially beneficial.

"Two-way data exchange ensures that our decision-makers have access to the most current and relevant information, allowing for informed and timely decision-making." - P5

6.7. Better Customer Experience

Bidirectional data pipelines guarantee that customer data is updated and consistent, which improves the customer experience. This makes it possible for businesses to provide their customers with services that are more responsive and customized.

"Better customer experience is the main advantage we gain with the two-way communication in our data pipelines." - P5

6.8. Flexibility and Adaptability

Bidirectional data pipelines offer the flexibility to react quickly to evolving data requirements and business needs. They enable businesses to modify their real-time data management procedures, provided that the data is still pertinent and helpful in continuously evolving business contexts.

"You still have to develop your solution to be flexible and adaptable because you can set up a bidirectional pipeline. That's not something that you can easily change what it's doing, what it's collecting, or what it's feeding back. The bidirectional pipelines facilitate more, you know, flexibility and adaptability, but that doesn't guarantee it just bringing a bidirectional pipeline doesn't give you that flexibility. It's a facilitator!" - P2

In summary, as shown in Fig. 8 bidirectional data pipelines offer organizations the advantages of real-time synchronization, better data management, improved decision-making, and improved customer experiences, making them essential tools for maintaining data integrity and streamlining operations in today's interconnected business environments.

7. Challenges of Bidirectional Data Pipelines (RQ2)

While bidirectional data pipelines offer significant advantages, they also present several challenges that organizations should be aware of when developing and maintaining these pipelines. Some of the key challenges of bidirectional data pipelines, based on our study, are the following:

7.1. Complex implementation and management

Bidirectional data pipeline implementation and management can be challenging, requiring careful configuration to guarantee smooth data flow in both directions. If this complexity is not adequately addressed, it may raise the possibility of mistakes and inefficiencies.

"I would say they're more complex, and I don't know how significantly more complex, but there's definitely an added complexity to it because now you're interacting back with a system in essentially like a right method instead of just reading something. So there's definitely added complexity, and that's going to mean that the level of complexity will be dependent on what the actual solution is. Beyond just reading data in a single direction, there's definitely more complexity to put there." - P1

7.2. Data Consistency and Integrity

It can be difficult to maintain data integrity and consistency across multiple systems, particularly when handling an extensive amount of data. Robust data management practices are necessary to ensure data accuracy and completeness while synchronizing data in real time between various systems.

"Especially if you're thinking of a multi-node system where you've got multiple sources and that all has to be aggregated somewhere via these bidirectional pipelines. How do you handle it when 11 nodes don't report? One node fails, and now you've got a consistency issue at the end of the pipeline where you know three of the four nodes reported their data and one is missing. How do you handle that? So it's definitely an added challenge, and it gets exacerbated if you have more. It's like you've got a cluster that you're, you know, synchronizing. Then you need to handle what happens when parts of it fail, not all of it. Everything fails. You use some kind of retry mechanism just like you would with a one-to-one relationship, but you have a many-to-one relationship. Now you've got to figure out how you do retries on individual failures and not systematic whole failures." - P5

7.3. Data Conflicts

When trying to update data simultaneously from multiple sources, bidirectional data pipelines may run into problems. It can be difficult and crucial to resolve these conflicts in a way that preserves the most recent and accurate data.

"Those consistency problems will result in a data conflict, yeah. Let's say you know you have A and B. You received a B update. An update failed, so you take the last value you received for A and make your decision or send something back based on that. Now you're going to have a conflict because something may have changed, but you're assuming it didn't. That's a conflict, so if you're making, if you're assuming the last value you did receive is still valid, you would run into a conflict. But if you're under the assumption that you didn't receive the value, you can't make a decision on that because you don't know. Now you have a consistency issue, but you don't have a conflict, right?" - P2

7.4. Security Risks

Sensitive data can be vulnerable to security risks when it is transmitted in both directions between systems. To reduce the risk of data breaches and unwanted access, strong security measures such as data encryption, access controls, and authentication protocols must be put in place.

"I believe that the security risks are almost similar for both unidirectional and bidirectional data pipelines. So a comprehensive security strategy that includes robust encryption, authentication measures, access controls, and continuous monitoring is indispensable for ensuring the safety and integrity of data." - P1

7.5. Performance and Latency Issues

Performance and latency problems can arise with bidirectional data pipelines, particularly when handling large data sets or requiring real-time data processing. One of the biggest challenges is making sure the pipeline can handle large amounts of data without compromising performance.

"It's also a challenge for unidirectional, but I would say that most of the time in a unidirectional pipeline you're not making some time-critical decision because you don't have a way to push that decision back or actuate anything. There might be cases where you do need to make a quick decision on the data coming in, and you're not doing some kind of feedback loop, but it's more prevalent if you have bidirectional, you're definitely sending something back and making some decisions, so speed is critical." - P3

7.6. Scalability Problems

It can be difficult to scale bidirectional data pipelines to meet increasing operational demands and rising data volumes. Careful planning and infrastructure investments are needed to ensure that the pipeline can handle growing data traffic and processing demands while preserving data consistency and integrity.

"One notable challenge is the potential for increased data volume and processing demands. The bidirectional nature of these pipelines means that not only are we extracting and analyzing data, but we're also integrating insights back into operational systems. As our organization grows and data sources multiply, there's a natural inclination for the bidirectional flow to intensify, putting a strain on the scalability of the infrastructure." - P1

7.7. Dependencies

Dependencies between various systems are typical when we use bidirectional data pipelines, making it difficult to update or modify one system without impacting others. Careful planning and coordination are needed to manage these dependencies and make sure that modifications in one system do not disrupt data flow in other linked systems.

"When you're just unidirectional, you're only dependent on the system being able to send you the information if something changes. With that, you've got to update. But now if you're pushing something back into it now you've got multiple interfaces that you're dependent on. So you have multiple things you have to keep track of. If they're not your system, and decides to change behavior of something, we have to keep up to date with that. And if we're using APIs to not only read stuff, but push stuff in, now we need to keep up with what they're changing on both sides of that. So yeah, there are more dependencies with bidirectional because now instead of just dependent on read operations or dependent on read and write into these systems" - P2

To address these challenges, as shown in fig. 8, organizations should implement robust data management practices, invest in reliable infrastructure, prioritize data security, and regularly monitor and optimize the performance of bidirectional data pipelines.

8. Steps to migrate from unidirectional to bidirectional data pipelines (RQ3)

Transitioning from unidirectional data pipelines to bidirectional data pipelines involves several key steps to ensure a smooth and successful migration process, as shown in fig 9. Here is a general guide for this transition developed based on our empirical study:

8.1. Evaluation and Planning

- Evaluate the architecture, data flow, and integration points of the existing data pipeline infrastructure.
- Determine which particular use cases and business procedures call for two-way data synchronization.
- Clearly define the goals and anticipated results when applying bidirectional data pipelines into practice.

8.2. Data Compatibility and Mapping

- Determine whether the target and source systems' data structures and formats are compatible.
- Map the fields and data items that require bidirectional synchronization in order to ensure integrity and consistency during the transition.

8.3. Infrastructure and Tools Selection

- Determine and pick the proper resources, including tools and infrastructure, to enable bidirectional data synchronization.
- It is advisable to employ resilient data integration platforms and technologies that enable smooth data transfer and instantaneous system synchronization.

8.4. Data Security and Governance

- Implement robust security measures to safeguard data integrity and confidentiality during the transition process.
- Establish data governance policies and protocols to ensure compliance with industry regulations and standards.

8.5. Configuration and Testing

- Configure the bidirectional data pipelines to ensure smooth data transfer and synchronization in both directions.
- Conduct comprehensive testing to verify the accuracy and reliability of data synchronization between the source and target systems.

8.6. Monitoring and Optimization

- Monitor the performance of the bidirectional data pipelines regularly to identify any potential issues.
- Optimize the pipelines based on the insights gained from monitoring, making necessary enhancements to improve data flow and synchronization.

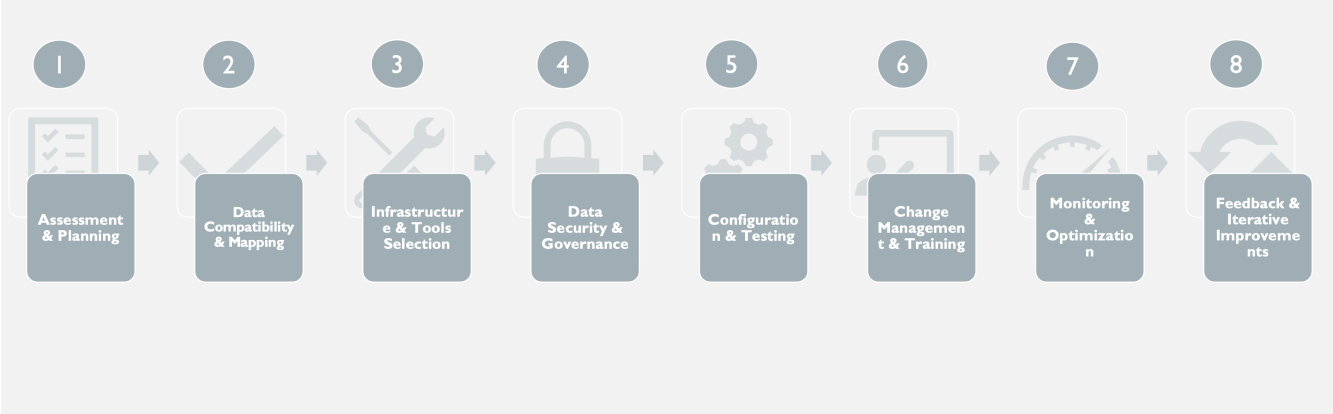

Figure 9: Steps to migrate from unidirectional to bidirectional data pipelines

8.7. Feedback and Iterative Improvements

- Collect feedback from end-users to assess the effectiveness of the transition to bidirectional data pipelines.
- Implement iterative improvements based on the feedback, ensuring that the bidirectional data pipelines continue to meet the evolving needs of the organization.

By following these steps, organizations can successfully transition from unidirectional data pipelines to bidirectional data pipelines, enabling seamless data synchronization and improved data management across interconnected systems and applications.

9. Conclusion and research implications

Data pipelines help organizations fully utilize their data for strategic decision-making, operational effectiveness, and preserving a competitive edge in the digital world. Bidirectional data pipelines present a dynamic and adaptable dimension to data management, providing a two-way flow of information between source and destination. The importance of bidirectional data pipelines lies in their potential to address challenges and requirements that unidirectional pipelines may not fully meet. However, there is a dearth of comprehensive studies focusing on bidirectional data pipelines. In this study, we have done a comprehensive investigation of unidirectional data pipelines without shared data transmission channels and bidirectional data pipelines, focusing on their necessity, benefits, challenges, differences from unidirectional data pipelines, and the migration process from unidirectional to bidirectional data pipelines. The need for bidirectional data pipelines has been identified as a result of evolving data requirements, wherein responsive and dynamic systems depend on a two-way data flow. The benefits outlined encompass real-time synchronization, efficient data management, improved decision-making, and enhanced customer experiences, making them essential tools for maintaining data integrity. However, the progression toward bidirectional pipelines has to overcome challenges such as data consistency and integrity, data conflicts, security concerns, and the need for robust error-handling mechanisms. The migration process from unidirectional to bidirectional pipelines involves careful examination of existing infrastructure, ensuring feasibility, necessity, compatibility, and minimizing disruptions. It requires thoughtful system design that includes protocol modifications, data synchronization, and the deployment of effective feedback loops. The research aims to provide practitioners with guidance on navigating the challenges involved in adopting bidirectional data pipelines as organizations become more aware of their benefits in promoting agility and responsiveness. By understanding the needs, benefits, challenges, differences, and transitional processes, practitioners can make informed decisions to harness the full potential of bidirectional data pipelines in the continuously evolving field of data management.

The implications for researchers are that more

study is required to maximize the performance of bidirectional pipelines, with an emphasis on sophisticated algorithms and architectures for higher efficiency. Our study also suggests examining the long-term impact of bidirectional pipelines, scalability factors, and adaptability to technological evolution. As bidirectional data flow becomes increasingly prevalent, researchers can delve into real-time anomaly detection within bidirectional pipelines. First and foremost, by weighing the benefits and challenges of bidirectional data pipelines against their organizational needs, practitioners can use the information covered in this paper to make well-informed decisions about switching from unidirectional to bidirectional data pipelines. Further, the findings help them position themselves in the transition stages and take steps to progress to the next level. Overall, this research equips practitioners with a nuanced understanding, empowering them to navigate the complexities of bidirectional data pipelines effectively within their organizational contexts.

10. Acknowledgement

During the preparation of this work, the author(s) used generative AI to improve language and readability. After using this tool or service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the content of the publication.

References

[1] N. Lakhan, Applications of data science and ai in business, International Journal for Research in Applied Science and Engineering Technology (2022). doi:doi: 10.22214/ijraset.2022.43343.

[2] F. Emmert-Streib, From the digital data revolution toward a digital society: Pervasiveness of artificial intelligence, Mach. Learn. Knowl. Extr. 3 (2020) 284–298. doi:doi: 10.3390/make3010014.

[3] W. Hurst, C. Dobbins, Guest editorial special issue on: Big data analytics in intelligent systems 3 (2015) 1–9. doi:doi: 10.12691/JCSA-3-3A-1.

[4] C. Mühlroth, M. Grottke, Artificial intelligence in innovation: How to spot emerging trends and technologies, IEEE Transactions on Engineering Management 69 (2020) 493–510. doi:doi: 10.1109/TEM.2020.2989214.

[5] A. R. Munappy, J. Bosch, H. H. Olsson, Data pipeline management in practice: Challenges and opportunities, in: Product-Focused Software Process Improvement: 21st International Conference, PROFES 2020, Turin, Italy, November 25–27, 2020, Proceedings 21, Springer, 2020, pp. 168–184.

[6] D. Wu, L. Zhu, X. Xu, S. Sakr, D. W. Sun, Q. Lu, Building pipelines for heterogeneous execution environments for big data processing, IEEE Software 33 (2016) 60–67. doi:doi: 10.1109/MS.2016.35.

[7] P. O'Donovan, K. Leahy, K. Bruton, D. T. O'Sullivan, An industrial big data pipeline for data-driven analytics maintenance applications in large-scale smart manufacturing facilities, Journal of big data 2 (2015) 1–26.

[8] K. Cottur, V. Gadad, Design and development of data pipelines, Int. Res. J. Eng. Technol.(IRJET) l0l0 7 (2020) 2715–2718.

[9] M. W. Van Alstyne, G. G. Parker, S. P. Choudary, Pipelines, platforms, and the new rules of strategy, Harvard business review 94 (2016) 54–62.

[10] A. R. Munappy, J. Bosch, H. H. Olsson, On the trade-off between robustness and complexity in data pipelines, in: Quality of Information and Communications Technology: 14th International Conference, QUATIC 2021, Algarve, Portugal, September 8–11, 2021, Proceedings 14, Springer, 2021, pp. 401–415.

[11] A. Freitas, E. Curry, Big data curation, New horizons for a data-driven economy: A roadmap for usage and exploitation of big data in Europe (2016) 87–118.

[12] A. Raj, J. Bosch, H. H. Olsson, T. J. Wang, Modelling data pipelines, in: 2020 46th Euromicro conference on software engineering and advanced applications (SEAA), IEEE, 2020, pp. 13–20.

[13] M. T. Spin Wang, Bidirectional Data Flow with Automated Decision-Making and Interactive Analytics, Technical Report, Tetrascience, ???? URL: https://assets-global.website-files.com/5fd7ad73fd90bb1e48d147c4/601b69cd2b27662a2d2d078d_Bidirectional%20White%20Paper.pdf.

[14] A. Dakkak, A. R. Munappy, J. Bosch, H. H. Olsson, Customer support in the era of continuous deployment: A software-intensive embedded systems case study, in: 2022 IEEE 46th Annual Computers, Software, and Applications Conference (COMPSAC), IEEE, 2022, pp. 914–923.

[15] M. Stach, C. Vogel, T.-C. Gablonski, S. Andreas, T. Probst, M. Reichert, M. Schickler, R. Pryss, Technical challenges of a mobile application supporting intersession processes in psychotherapy, Procedia Computer Science 175 (2020) 261–268.

[16] A. P. McAfee, E. Brynjolfsson, Big data: the management revolution., Harvard business review 90 10 (2012) 60–6, 68, 128.

[17] F.-Z. Benjelloun, A. A. Lahcen, S. Belfkih, An overview of big data opportunities, applications and tools, 2015 Intelligent Systems and Computer Vision (ISCV) (2015) 1–6. doi:doi: 10.1109/ISACV.2015.7105553.

[18] P. Chandarana, M. Vijayalakshmi, Big data analytics frameworks, 2014 International Conference on Circuits, Systems, Communication and Information Technology Applications (CSCITA) (2014) 430–434. doi:doi: 10.1109/CSCITA.2014.6839299.

[19] T. Kosar, G. Kola, M. Livny, Data pipelines: enabling large scale multi-protocol data transfers (2004) 63–68. doi:doi: 10.1145/1028493.1028504.

[20] D. Roman, N. Nikolov, A. Soylu, B. Elvesæter, H. Song, R.-C. Prodan, D. Kimovski, A. Marrella, F. Leotta, M. Matskin, G. Ledakis, K. Theodosiou, A. Simonet-Boulogne, F. Perales, E. Kharlamov, A. Ulisses,

A. Solberg, R. Ceccarelli, Big data pipelines on the computing continuum: Ecosystem and use cases overview, 2021 IEEE Symposium on Computers and Communications (ISCC) (2021) 1–4. doi:doi: 10.1109/ISCC53001.2021.9631410.

[21] A. Munappy, D. I. Mattos, J. Bosch, H. H. Olsson, A. Dakkak, From ad-hoc data analytics to dataops, Proceedings of the International Conference on Software and System Processes (2020). doi:doi: 10.1145/3379177.3388909.

[22] H. V. Alphen, C. Avezaat, R. Thomeer, Preface, Clinical Neurology and Neurosurgery 99 (1997). doi:doi: 10.1016/S0303-8467(97)81249-8.

[23] K. Rengarajan, V. Menon, Generalizing streaming pipeline design for big data (2019) 149–160. doi:doi: 10.1007/978-981-15-1366-4_12.

[24] B. Varghese, N. Wang, S. Barbhuiya, P. Kilpatrick, D. S. Nikolopoulos, Challenges and opportunities in edge computing, 2016 IEEE International Conference on Smart Cloud (SmartCloud) (2016) 20–26. doi:doi: 10.1109/SmartCloud.2016.18.

[25] M. Aiswarya Raj, J. Bosch, H. H. Olsson, A. Jansson, On the impact of ml use cases on industrial data pipelines, in: 2021 28th Asia-Pacific Software Engineering Conference (APSEC), 2021, pp. 463–472. doi:doi: 10.1109/APSEC53868.2021.00053.

[26] M. Aiswarya Raj, J. Bosch, H. H. Olsson, Maturity assessment model for industrial data pipelines, 2023. Unpublished manuscript.

[27] O. Oleghe, K. Salonitis, A framework for designing data pipelines for manufacturing systems, Procedia CIRP 93 (2020) 724–729. doi:doi: 10.1016/j.procir.2020.04.016.

[28] A. Dakkak, H. Zhang, D. I. Mattos, J. Bosch, H. H. Olsson, Towards continuous data collection from in-service products: Exploring the relation between data dimensions and collection challenges, in: 2021 28th Asia-Pacific Software Engineering Conference (APSEC), IEEE, 2021, pp. 243–252.

[29] R. E. Stake, Multiple case study analysis, Guilford press, 2013.

[30] J. M. Verner, J. Sampson, V. Tosic, N. A. Bakar, B. A. Kitchenham, Guidelines for industrially-based multiple case studies in software engineering, in: 2009 Third International Conference on Research Challenges in Information Science, IEEE, 2009, pp. 313–324.

[31] R. K. Yin, Case study research design and methods third edition, Applied social research methods series 5 (2003).

[32] P. Baxter, S. Jack, et al., Qualitative case study methodology: Study design and implementation for novice researchers, The qualitative report 13 (2008) 544–559.

[33] P. Runeson, M. Höst, Guidelines for conducting and reporting case study research in software engineering, Empirical software engineering 14 (2009) 131.

[34] V. Clarke, V. Braun, N. Hayfield, Thematic analysis, Qualitative psychology: A practical guide to research methods 3 (2015) 222–248.