Toward a DataOps Framework for Enhancing Data Quality in Data Science

Christian Haertel
Otto-von-Guericke-University
Magdeburg, Germany *christian.haertel@ovgu.de*

Kunal Sanjay Sagavakar
Otto-von-Guericke-University
Magdeburg, Germany *kunal.sagavakar@ovgu.de*

Daniel Staegemann
Otto-von-Guericke-University
Magdeburg, Germany *daniel.staegemann@ovgu.de*

Matthias Pohl
German Aerospace Center (DLR) - Institute of Data Science
Jena, Germany *matthias.pohl@dlr.de*

Matthias Volk
T-Systems International GmbH
Magdeburg, Germany *M.Volk@t-systems.com*

Klaus Turowski
Otto-von-Guericke-University
Magdeburg, Germany *klaus.turowski@ovgu.de*

Abstract

Data Science (DS) leverages Data Analytics to help organizations extract value from large datasets and enhance performance. A significant focus in DS is on data collection, cleaning, and transformation to create high-quality datasets for analysis. However, traditional manual data preparation methods are often inefficient and error-prone, particularly in Big Data environments. DataOps seeks to automate data lifecycle stages by integrating DevOps practices, enhancing the quality and reliability of data pipelines. This paper proposes a framework for implementing DataOps, demonstrated through a case study on urban mobility analytics.

Keywords: Data Science, Data Analytics, DataOps, Design Science, Cloud Computing.

1. Introduction

Organizations increasingly rely on Data Analytics (DA) to support strategic decision-making in a data-driven world. The rise of Big Data, characterized by high volume, velocity, and variety, has drawn attention to the potential of extracting value from data. DA plays a vital role in the Data Science (DS) lifecycle, yet the growing complexity and volume of data lead to challenges in effective management and analysis [4]. Many organizations struggle to realize the value of their data assets due to a lack of methodologies that accommodate the dynamic nature of DS, resulting in fragmented workflows and delays in implementing analytics solutions. Traditional data processing methods, relying on manual scripting, fall short of meeting demands for real-time analytics and scalability, limiting organizational agility [6]. The absence of automation and orchestration creates inefficiencies and increases errors, while managing data across multiple

platforms compounds these issues [16]. DataOps emerges as a promising solution, bringing DevOps principles to DA [16]. It aims to enhance collaboration, automate deployment, and improve the quality and speed of analytics [12]. However, despite its potential, there is a lack of methodological guidance and demonstrations in the literature [8]. Thus, this research proposes a DataOps framework grounded in established core principles, employing the Design Science Research (DSR) methodology [9], and addressing the following research question (RQ):

RQ: *How can a DataOps framework for Data Science be designed and implemented?*

The paper at hand discusses the theoretical fundamentals and related work on DataOps. A DataOps framework is introduced, and its application for urban mobility is presented. The work concludes by summarizing findings and suggesting future research directions.

2. Theoretical Background and Related Work

DS aims to derive actionable insights from large datasets using scientific methods. It is closely linked to DA, which systematically collects, cleans, and interprets data to identify patterns and trends for informed decision-making [12]. DA is a subprocess of the DS lifecycle, which includes stages such as Business Understanding, Data Collection and Preparation, Analysis, Evaluation, Deployment, and Utilization [4], [7]. Despite the potential of DA, many projects fail due to challenges like low data quality and lack of reproducibility [12], [18]. Thus, new approaches are necessary to tackle these issues, particularly in data operations where significant effort revolves around data transformation and cleaning [2], [5], [15], [17]. DataOps emerges as a solution, combining practices from agile software engineering and DevOps to automate and orchestrate data lifecycle stages for consistent, high-quality results [13]. Key DataOps principles include automation, continuous integration (CI) and continuous deployment (CD), version control, and quality monitoring, all of which facilitate collaboration and transparency. While DataOps can enhance DS initiatives, academic literature on its practical application is limited. Notable exceptions exist, such as use cases in cyber-physical systems and superficial conceptual frameworks [6], [13], [16]. Addressing these research gaps, a comprehensive DataOps framework, embedded in the DS lifecycle, is proposed.

3. Conceptual Design of the DataOps Framework

This section describes the design of the DataOps framework aligned with established DataOps principles. Traditional data processing methods often struggle with efficiency and agility in dynamic environments, especially for Big Data analytics. The proposed framework addresses these challenges by incorporating automation in data management, testing, version control, metadata management, quality assurance, environment separation, monitoring, and error handling. It draws on the relevant literature to provide analytics teams with high-quality curated data [3], [6], [10, 11], [13], [16]. Illustrated in Fig. 1, the framework follows a lifecycle model that ensures a separation between the development and production environments, allowing experimentation without disrupting business operations. Business objectives and analytical goals guide the design of the system, determining the necessary data sources. After initial exploration, the required data transformations are identified to tackle quality issues and create relevant features for analytics. A source code repository facilitates version control and developer collaboration [10]. The deployment of results is enhanced by automated CI/CD tools that help catch issues early, preventing defects in the production environment. A comprehensive test suite ensures performance and reliability at every pipeline stage, establishing data quality according to requirements [13]. Unit tests verify data transformations for correctness, while integration tests validate the entire data flow from ingestion to output. Performance testing assesses scalability and resource utilization under peak conditions. Artifacts that contain the data pipeline are created and stored [10]. The orchestration component automates task triggering based on busi-

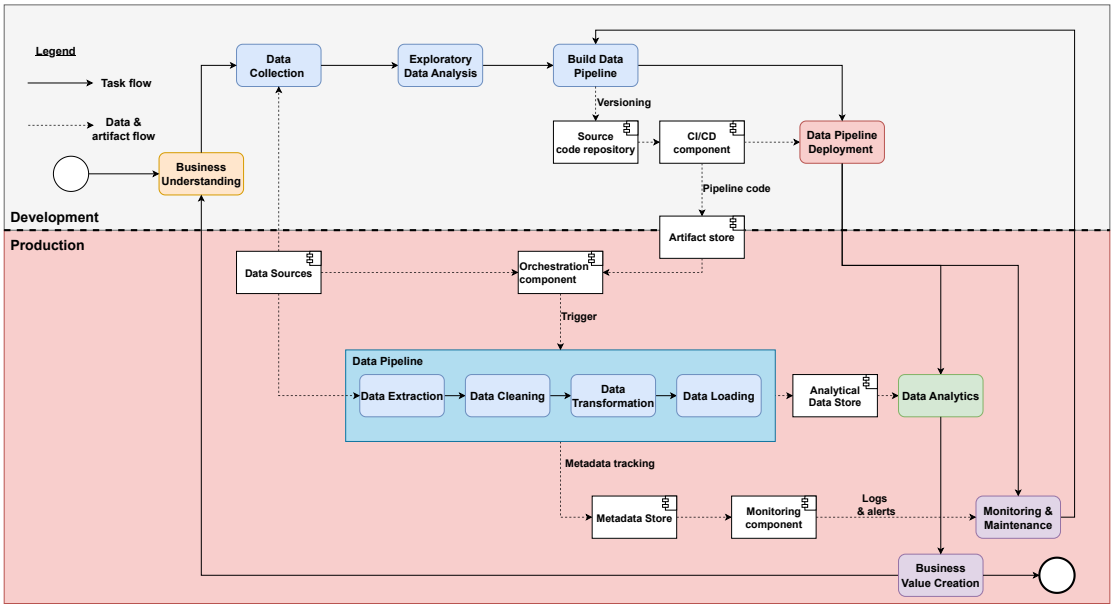

Fig. 1. DataOps Framework

ness requirements, while the DataOps approach accelerates data insight delivery and enhances quality through predefined inputs and outputs for each stage. Version control and metadata management track dataset versions, enabling dependency tracking and data lineage. In production, maintaining high data quality and minimizing downtime are vital. A feedback system using key performance indicators supports adaptive adjustments [13]. Continuous monitoring of data processing helps detect issues, while metrics like latency and throughput track pipeline performance. Error logs facilitate trend analysis and compliance audits, providing insights for improvements. Automated alerts can address anomalies swiftly. If modifications are needed, the framework allows reverting to the pipeline-building step, promoting agility and reliability. The data pipeline ultimately loads processed data into an analytical data storage. The subsequent DA phase is abstract, varying by business objectives, and can either continue or restart the process if objectives change.

4. Case Study

The applicability of the presented framework is demonstrated on a use case in urban mobility, where efficient transportation solutions in smart cities are needed. The integration of DataOps practices can enhance mobility data management and improve decision-making. For example, the transportation network of Lisbon struggled with ticketing data due to its volume and heterogeneity, necessitating advanced processing techniques for real-time insights [1]. This study uses the New York City (NYC) Taxi dataset as an example of urban mobility data. Although only a subset of the complete dataset is analyzed, it serves as a proof-of-concept for implementing the DataOps framework. The data, sourced from the NYC Taxi and Limousine Commission [14], spans two months (January and February 2017) and includes over two million trip records in a 1.7 GB CSV file, containing trip details like timestamps, locations, passenger counts, and fare information. The implementation follows the DataOps framework and is divided into two environments. The goal is to analyze taxi vendor performance and city locations, starting with data exploration in a development environment. The data is stored in Google Cloud Storage (GCS) and analyzed using BigQuery and Looker Studio. The data pipeline is orchestrated using directed acyclic graphs (DAGs) with Apache Airflow, integrated with Google Cloud's Cloud Composer for automation and monitoring. Airflow is paired with Docker to ensure reproducibil-

ity.

The source code is versioned in GitHub, with a GitHub Actions workflow triggering CI/CD via Cloud Build for automated testing and deployment. Airflow DAGs are containerized and stored in the Container Registry, which manages the images. The production environment utilizes Cloud Composer, hosting the three main DAGs. The data processing DAG includes extraction, validation, cleaning, transformation, and loading stages. Schema compliance is verified before ingesting data from the GCS bucket, and invalid records are removed to maintain integrity. Pickup and drop-off timestamps are used to calculate trip durations, with each event logged in the Cloud Composer and Airflow UI for traceability. Custom transformations calculate features such as total revenue and average fares for vendors and locations, with two final datasets loaded into BigQuery for further analysis and visualizations in Looker Studio. The other DAGs are employed for tracking metadata and monitoring, supplemented by Google Cloud Monitoring and Logging to enable pipeline performance tracking and error detection. Custom dashboards display metrics like task success rate, data quality errors, pipeline uptime, and resource usage. The autoscaling capability of Cloud Composer manages workload spikes, and the modular architecture supports enhancements like real-time data processing with Google Cloud PubSub for efficient streaming data ingestion.

5. Conclusion

A significant portion of DS projects focuses on data engineering and processing to ensure high data quality for analytics, necessitating an efficient and reproducible approach [2], [5], [15]. This paper investigates the design and implementation of a DataOps framework, emphasizing automation and quality assurance. We demonstrate the applicability of the framework using an urban mobility use case. The prototype utilizes cloud technologies like GCS and BigQuery, along with Airflow for workflow orchestration, and GitHub Actions for CI/CD. To improve the generalizability and utility of the framework, we suggest extending the evaluation to further cases and including a comparative analysis to traditional approaches using quantitative metrics to assess whether the intended improvements are achieved. Additionally, supplementing the artifact with MLOps principles can support the development and operationalization of DA [10].

References

[1] Antunes, H., Figueiras, P., Costa, R., Teixeira, J., Jardim-Goncalves, R.: Analysing Public Transport data through the use of Big Data technologies for urban mobility. 2019 International Young Engineers Forum (YEF-ECE) (2019)

[2] de Bie, T., de Raedt, L., Hernández-Orallo, J., Hoos, H.H., Smyth, P., Williams, C.K.I.: Automating Data Science. Communications of the ACM 65(3), pp. 76–87 (2022)

[3] Capizzi, A., Distefano, S., Mazzara, M.: From DevOps to DevDataOps: Data Management in DevOps Processes. In: Software Engineering Aspects of Continuous Development and New Paradigms of Software Production and Deployment, pp. 52–62. Lecture Notes in Computer Science, Springer International Publishing, Cham (2020)

[4] Chang, W.L., Grady, N.: NIST Big Data Interoperability Framework: Volume 1, Definitions. NIST Special Publication 1500-1r2 (2019)

[5] Dasu, T., Johnson, T.: Exploratory data mining and data cleaning. John Wiley & Sons (2003)

[6] Ereth, J.: DataOps - Towards a Definition. Proceedings of the Conference "Lernen, Wissen, Daten, Analysen" (2018)

[7] Haertel, C., Pohl, M., Nahhas, A., Staegemann, D., Turowski, K.: Toward A Lifecycle for Data Science: A Literature Review of Data Science Process Models. PACIS 2022 Proceedings (2022), https://aisel.aisnet.org/pacis2022/242

[8] Haertel, C., Staegemann, D., Daase, C., Pohl, M., Nahhas, A., Turowski, K.: MLOps in Data Science Projects: A Review. 2023 IEEE International Conference on Big Data (BigData) pp. 2396–2404 (2023)

[9] Hevner, A.R., March, S.T., Park, J.: Design Science in Information Systems Research. MIS Quarterly Vol. 28(No. 1), pp. 75–106 (2004)

[10] Kreuzberger, D., Kühl, N., Hirschl, S.: Machine Learning Operations (MLOps): Overview, Definition, and Architecture. IEEE Access 11, pp. 31866–31879 (2023)

[11] Mainali, K., Ehrlinger, L., Himmelbauer, J., Matskin, M.: Discovering dataops: A comprehensive review of definitions, use cases, and tools. DATA ANALYTICS 2021: The Tenth International Conference on Data Analytics (2021)

[12] Martinez, I., Viles, E., Olaizola, I.G.: Data Science Methodologies: Current Challenges and Future Approaches. Big Data Research 24 (2021)

[13] Munappy, A.R., Mattos, D.I., Bosch, J., Olsson, H.H., Dakkak, A.: From Ad-Hoc Data Analytics to DataOps. In: Proceedings of the International Conference on Software and System Processes. pp. 165–174. ACM, New York, NY, USA (2020)

[14] NYC Taxi & Limousine Commission: TLC Trip Record Data (2025), https://www.nyc.gov/site/tlc/about/tlc-trip-record-data.page

[15] Paton, N.W.: Automating Data Preparation: Can We? Should We? Must We? Workshop Proceedings of the EDBT/ICDT 2019 Joint Conference (2019)

[16] Rodriguez, M., de Araújo, L.J.P., Mazzara, M.: Good practices for the adoption of DataOps in the software industry. Journal of Physics: Conference Series 1694(1) (2020)

[17] Saltz, J.S., Krasteva, I.: Current approaches for executing big data science projects - a systematic literature review. PeerJ Computer Science 8(e862) (2022)

[18] VentureBeat: Why do 87% of data science projects never make it into production? (2019), https://venturebeat.com/2019/07/19/why-do-87-of-data-science-projects-never-make-it-into-production/