Towards Industry 5.0: AAS/MLOps-Driven Model Maintenance for Data-Centric Production*

Kiavash Fathi[1,2], Marcin Sadurski[2], Stefan Waskow[3],
Tobias Kleinert[1] and Hans Wernher van de Venn[2]

[1]*Chair of Information and Automation Systems for Process and Material Technology, RWTH Aachen University, 52064 Aachen, Germany*

[2]*Institute of Mechatronic Systems, Zurich University of Applied Sciences, 8400 Winterthur, Switzerland*

[3]*OEE Improvement team, ABB, 8200 Schaffhausen, Switzerland*

Keywords: Model and Data Life-Cycle, Machine Learning Operations (MLOps), Asset Administration Shell (AAS), Human-in-the-Loop Design, Industry 5.0.

Abstract: Despite the advancements brought by digitalization across industries, only a few state-of-the-art data-driven methods successfully transition to production and remain viable. The sheer volume of physical assets in production lines, combined with constantly evolving requirements, makes model deployment and maintenance highly complex. This paper presents a production-ready architecture developed for data-driven digital assets at ABB Schaffhausen AG. The solution integrates MLOps best practices orchestrated via MLRun with the industry-standard metadata modeling system, Asset Administration Shell (AAS). We demonstrate how controlled artifact generation from MLRun facilitates experiment tracking and knowledge sharing while AAS ensures standardization and long-term maintenance. By combining MLOps and AAS, we effectively manage the ever-growing artifacts of data-driven solutions. Additionally, we explore how controlled artifact generation enables role-based MLOps by restricting access to relevant information based on industrial roles. This architecture supports a smooth transition to Industry 5.0.

1 INTRODUCTION

One of the main focuses of production lines compliant with Industry 4.0 standards is **flexibility in production** which is normally implemented and deployed as modular automation. Due to the short lifespan of products and manufacturing technologies, modular automation helps ensure that final products meet constantly evolving customer requirements with minimal engineering effort. (Xia et al., 2023). Additionally, it makes sure that the production line is capable of coping with unforeseen situations in a safe and secure manner (Huang et al., 2021). This flexibility and adaptability will not only ensure fully personalized product for the market but also helps to achieve a more sustainable production (Ghobakhloo, 2020; Brettel et al., 2016).

The demand for production flexibility has driven extensive research into the management and orchestration of physical and digital assets (DAs) in smart factories. In fact, deploying modular automation requires not only the integration of different assets, such as robots and automation systems (Müller et al., 2021c; Müller et al., 2021b), but also reconfigurable assets and processes (Müller et al., 2021a; Kozma et al., 2019).

All the changes induced in the physical assets is reflected in the data readings from them, ultimately impacting the related DAs. In that regard, given the fact that majority of the DAs implemented in Industry 4.0 settings for production monitoring and optimization are data-driven, there have been numerous research on data handling and data-driven model adaptation given changes in the production and processes. (Further details can be found in (Polke et al., 2023; Yue and Wang, 2022; Morgan et al., 2021)).

In all the above-mentioned research, one of the main concerns during the solution development is changes in the data exposed during data-driven model

*This project is part of the research conducted at Competence Centre for Automation and Digitalisation entitled Pre-Prime (Pre-Process Refinement for Improved Manufacturing Efficiency) in collaboration with ABB Schaffhausen AG.

Fathi, K., Sadurski, M., Waskow, S., Kleinert, T. and van de Venn, H. W.
Towards Industry 5.0: AAS/MLOps-Driven Model Maintenance for Data-Centric Production.
DOI: 10.5220/0013716300003982

In *Proceedings of the 22nd International Conference on Informatics in Control, Automation and Robotics (ICINCO 2025) - Volume 2*, pages 495-502
ISBN: 978-989-758-770-2; ISSN: 2184-2809

training compared to model deployment and testing data. This discrepancy between the two data distributions, and also changes in the data in general, is an active research area in the field of Machine Learning Operations (MLOps).

As the matter of fact, MLOps aims to facilitate the efficient deployment and serving of machine learning (ML) models in production which consequently also constantly improves the business activities (Haviv and Gift, 2023). In that regard, for tracking changes in the production line and adapting the DAs, it is crucial to **version** the data, models and also the code. Doing so makes the entire pipeline repeatable and traceable for performance evaluation and comparison (Faubel et al., 2023). To this regards, we aim to answer the following research question (RQ) concerning model serving:

> **RQ1:** How can an orchestration tool like MLRun **efficiently**[1] track experiments, data, and models in artifact form for knowledge sharing in data-centric production?

In what follows, we use predictive maintenance (PdM) as an example to explain the deployment related problems in industry. Nonetheless, the proposed architecture can be generalized to any other data-driven solution such as zero defect manufacturing (ZDM), CO_2 emission tracking (CO2-T), *etc*.

In the field of PdM, as an important concept in Industry 4.0, research dealing with MLOps tries to ensure the performance of deployed solution despite the changes in the production line. Such solutions suggest to have triggers for retraining the data-driven model either given a drop in the accuracy, or other performance indicators (Raj et al., 2021; Fathi et al., 2024b), or instead doing it periodically (Oluyisola et al., 2020; Shakhovska and Campos, 2024).

There are several issues for model maintenance using the aforementioned approaches. Firstly, the lack of annotated data in PdM is a well-known but still unresolved issue. Thus, accuracy-based triggers are inefficient, as model adaptation occurs too late—only after downtime has already occurred. Secondly, these approaches neglect the operator's domain knowledge for maintaining a physical asset which is counter productive. Numerous successful production lines have operators which have years of experience working with different production assets. Therefore, it is of utmost importance to integrate their knowledge into the MLOps design and deployment cycle. Lastly and most importantly, **high model accuracy** does not automatically translate into **more profit, increased sustainability, *etc.***, for the company and rather the impact of the deployed model should be monitored and evaluated by the plant or asset operator (Haviv and Gift, 2023).

Unfortunately, MLOps studies trying to include the business side's opinion and/or expertise into the design and deployment, when at all, merely consider their expertise only in the initial concept development (Faubel et al., 2023; Colombi et al., 2024; Kreuzberger et al., 2023; Salama et al., 2021). However, we believe a more effective solution would be to integrate the asset operator in the maintenance of the PdM models as well. This will allow the operator to make requests to the ML engineer directly (Fathi et al., 2024a). Fig. 1 shows our proposal for this integration which is heavily based on the work from Faubel *et al.* (Faubel et al., 2023). The arrows in this figure, indicate the actions performed by the roles within the dashed rectangles to reach the next step. Furthermore, each dashed rectangle describes the contributions of these roles in the corresponding MLOps step. With this solution we aim to answer the following RQ:

> **RQ2:** How can domain knowledge from the shop floor be **effectively**[2] integrated with artifacts generated in data-driven architectures?

In that regard, another important improvement in our proposed solution for maintaining data-driven models is the integration of different required roles. As a generic and high-level solution, instead of breaking down the roles needed for developing and deploying a data-driven solution given MLOps requirements (Colombi et al., 2024; Kreuzberger et al., 2023), namely data scientist, data engineer, *etc.*, we divide them in the following ones (refer to Section 3 for more details):

1. Operator
2. ML engineer
3. Ecosystem manager

The main reason for defining the roles as suggested above is **effective information sharing** between different actors working in a company on a specific product or a production asset (Milicic et al., 2016).

After **integrating domain knowledge** in the proposed solution and also **defining roles** the last and most important step would be to choose a **represen-**

[1] **Efficient** in terms of reducing excessive overhead generation and preventing bursts of excessive information.

[2] **Effective** in terms of providing added value and semantic meaning (Zhou et al., 2023) to different parts of a pipeline.

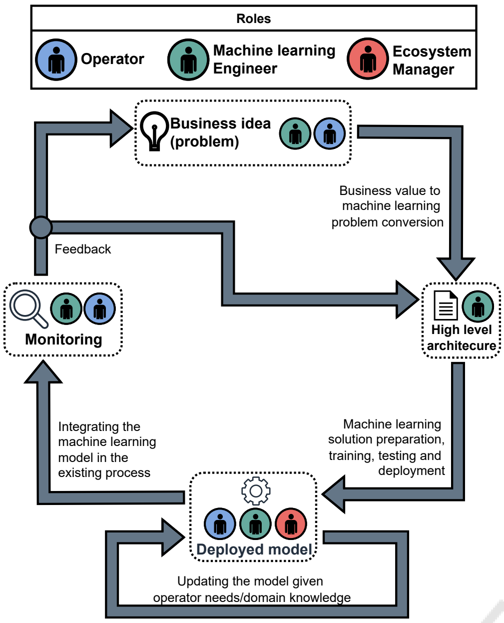

Figure 1: Overview of the DA maintenance cycle (refer to Section 3 for detailed role descriptions).

tation for sharing information between different actors. This leads to the following RQ:

> **RQ3:** How can information from data-driven solutions be structured in an industry-approved format?

For addressing this issue, we utilize the Asset Administration Shell (AAS) (Wei et al., 2019) as a standardized digital representation of digital and physical assets in industry.

These steps will enable successful product life cycle management in industrial use cases, where product is the deployed DA in smart factories.

Lastly, as a closing statement of this section, by highlighting the roles in the proposed ecosystem, we aim to shift the current model-driven design towards a product-oriented one as suggested by Kreuzberger *et al.* (Kreuzberger et al., 2023). Only then it is possible to implement the concepts required by Industry 5.0 for bringing human in model maintenance loop and increasing the acceptance of models in the modern and complex industrial plants. As the final goal of this paper we hope to pave the way for a balance between **automatization** and **collaboration** as introduced by (Ruppert et al., 2022; Sabuncu and Bilgehan, 2025).

As a short summary, our main contributions are thus 3-fold. We:

1. pinpoint the importance of efficient artifact generation,
2. provide a systematic way for coupling information from shopfloor to the developed and deployed DAs (and their artifacts),
3. propose an industry approved system to handle artifacts from MLRun.

2 PROBLEM SETTING AND FOCUS AREA

During the development of this solution, we focused heavily on two distinctive aspects of production lines. Namely, the **lack of (annotated) historical data** and therefore the importance of human feedback and also the overwhelming **number of physical assets** to be monitored. The former results in excessive number of (experimental) implementations with their corresponding artifacts containing data, metadata, models, *etc.*, which need to be versioned for improved traceability. The later further exacerbates the problem by extending this issue across multiple physical assets, whose performance is decisive in the productivity of the production line.

On top of that, various aspects of the production potentially require dedicated models for monitoring for enhancing their performance. These aspects include but are not limited to PdM, ZDM, CO2-T, *etc.*, which we collectively refer to as tasks in this paper. As a result, these monitoring systems will also lead to more artifact generation from the solution. Fig. 2, from a machine and not component point of view, shows how the number of artifacts can exponentially grow given the scale of the production line.

We aim to highlight the pivotal role of managing data and metadata generated by DAs. In fact, in Industry 4.0/5.0, physical assets are not the only type of assets that firstly **produce data** and secondly **require maintenance** in response to changes occurring in the production line.

In addition, given the current digitalization state of different industries, our observations are mostly in Switzerland, it is rarely the case that the available data is annotated for solving the optimization problem at hand. Therefore, the heavily focused automation in MLOps for model training and serving, *e.g.*, using AutoML, becomes obsolete. Please refer to (Salama et al., 2021; Colombi et al., 2024; Cerquitelli et al., 2021). This characteristic highlights the importance of implementation and artifact tracking and also human feedback for ensuring the performance of the system. After defining the boundaries for the problem at hand, in what follows, the architecture of the proposed method is introduced.

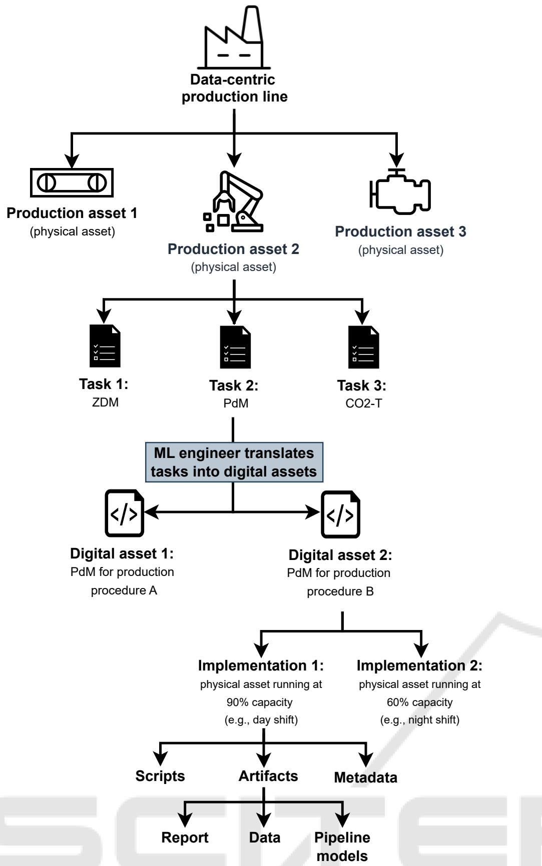

Figure 2: Exponential growth in artifact generation in data-centric production.

3 SYSTEM ARCHITECTURE

In industrial settings, model verification and deployment are inherently experimental due to the scarcity of annotated data. High model accuracy does not necessarily correlate with improved productivity or sustainability, making human involvement critical.

To address these challenges, our proposed system provides a foundation for generating and archiving DA implementations, along with corresponding information from human operators. This information provides sentiment for the generated DA. This in the long run facilitates DA tracking and matching developed for different physical assets or tasks. Our proposed architecture (Fig. 3) integrates three key roles:

1. **Operator:** Defines tasks and provides feedback.
2. **ML Engineer:** Develops and maintains DAs.
3. **Ecosystem Manager:** Oversees data aggregation and solution inspection.

The system workflow is as follows. Operators define tasks requiring DA development. Afterwards, ML engineers develop solutions and generate artifacts using MLRun.

Thereafter, the generated artifacts (data, metadata, models and their hyperparameters, *etc.*) are encapsulated within AAS submodel instances along with the task description. These artifacts contain the **minimum information** required to recreate the entire pipeline for the solution. In fact, despite the convenience of tools like MLRun and MLFlow, excessive artifact generation can make it difficult to organize and use this information effectively for decision-making. Lastly, the ecosystem manager inspects and manages DAs across production lines. Ultimately, this structure ensures traceability, effective feedback integration, and reduced model development costs.

For adapting and/or improving the deployed DA, the operator and the ecosystem manager can actively provide feedback to the ML engineer for further adjustment of the DA leading to new instances from these DAs in the database. Human agency is a key design choice in the proposed system. It is therefore assumed that provided feedback is deliberate, accurate and aligns with production requirements.

In what follows, our proposed structure for generating DA implementations in ABB Schaffhausen is provided. Please note that this structure is referred to as **Pre-Prime Ecosystem**.

4 INDUSTRIAL USE CASE: ABB PLANT

The Pre-Prime ecosystem developed for ABB Schaffhausen AG is shown in Fig. 4. Once the operator defines a new task for the ML engineer, they can use the provided Jupyter notebooks to work on different parts of the ML project. Given the requirements from ABB, we divided these Jupyter notebooks into the following:

1. **Data preparation and preprocessing:** contains all the operations needed for making the data vector ready for different algorithms.
2. **Model training and validation:** contains all the steps to go from an idea for solving the task to a verified and functional model(s).
3. **Model serving:** contains the logic for running the model and integrating it into the existing IT infrastructure of the target company.

However, these can change given the needs and/or limitations.

In addition, the parameters, hyperparameters, *etc.*, impacting the behavior of the implementation are also stored separately as a metadata file. Once the solution

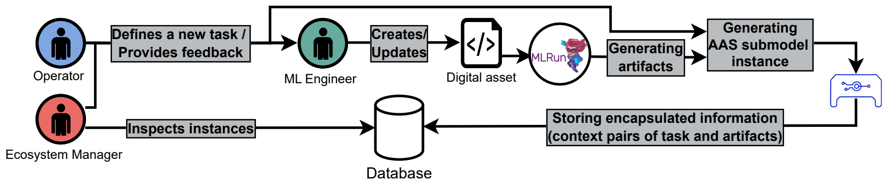

Figure 3: System overview.

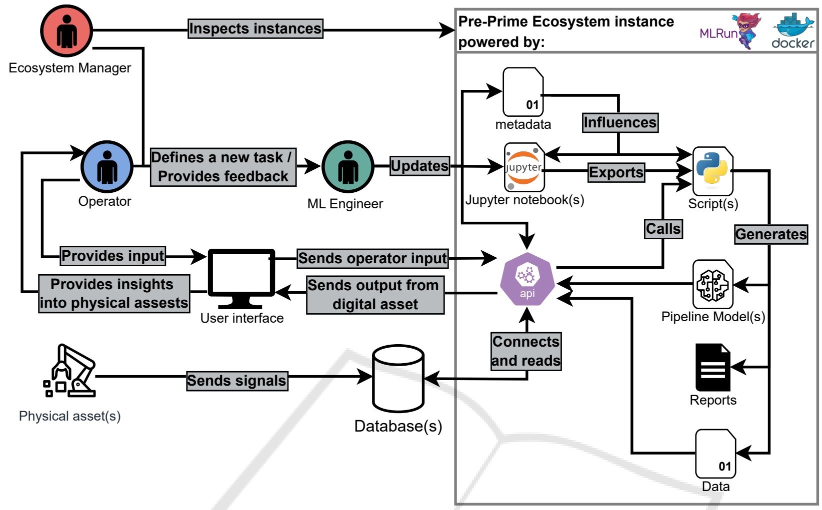

Figure 4: Overview of Pre-Prime ecosystem. Based on feedback from the ecosystem manager or operator, the ML engineer designs a new DA in Jupyter notebooks, generating updated scripts, models, reports, and data. These newly created artifacts are then integrated into the API during model serving to meet shopfloor requirements.

has reached the maturity level to be tested in the production line, scripts, models, reports and data (specifically used for model training, testing and verification) are extracted as a form of artifact. For the sake of traceability, the data before and after being vectorized are stored separately, so that in case of erroneous predictions, the pipeline can be debugged easier. This separation is also in accordance with the demands of the EU AI Act for safety critical systems (ISACA, 2024; Dorigo et al., 2025).

Later on, the designated API uses these artifacts to serve the operator and the rest of the implementation is frozen. The API basically contains all the endpoints for its different functionalities. Thereafter, in case the performance of the DA implementation is not acceptable, given the feedback from the operator and/or the ecosystem manager, the ML engineer can go thorough the development cycle and update the solution.

5 INTEGRATION WITH AAS

For having the most efficient and easily standardizable solution for industry, in this section we demonstrates how a data-centric production lines can leverage various official AAS submodels to optimize its operations.

As the main motivation, we aim to use the available AAS submodels for different physical and DAs to **ease communication** between them. As an example, an OEM provides submodels for their assets, which as a result help different customers access information from the assets in a unified way.

As shown in Fig. 5, the information about the physical asset is contained within the digital nameplate. This information is useful when the operator is defining a new task for which a DA is to be implemented.

Furthermore, for ensuring structural and schematic consistency of the data, we propose to use the time series data submodel. In fact, this submodel can also be used to detect potential data shifts impacting DA which are developed later.

In addition, the (data-driven) DAs are in accordance with the AI submodel[3] to ensure all the required information for model development, deployment and serving are contained.

[3]As of the writing this paper, this submodel is not yet published.

As a side note, the data-driven DAs can potentially benefit from other submodels as well. As of the writing this paper, these submodels include but are not limited to Carbon footprint, Predictive maintenance, *etc*. In fact, these submodels help the ML engineer incorporate the required domain knowledge, enhancing traceability and making troubleshooting easier for the solution.

Lastly, all these submodel instances along with their artifacts are encapsulated into the data-driven model maintenance submodel instance. In addition, the task description from the operator is also included in this submodel to provide the required sentiment to the implementation. The ecosystem manager then by looking at different instances of our proposed submodel can have all the required information for solution inspection, model and data aggregation, *etc*. Moreover, information from this submodel then can be filtered according to different access permissions of different roles, adding even more value to this submodel.

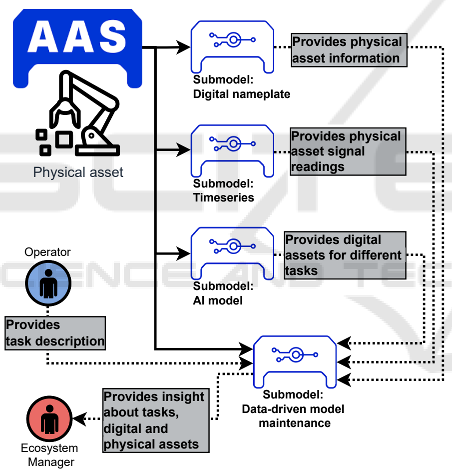

Figure 5: Physical and DA management via human-in-the-loop AAS.

6 DISCUSSION

In this paper, the foundation required for DA lifecycle management was presented. Such a solution can later be adapted for creating digital product pass (Psarommatis and May, 2024) for DAs in an Industry 4.0/5.0 setting. Furthermore, with the increasing use of advanced models, such as large language models (LLMs), traceability becomes a vital characteristic of AI-powered solutions. In fact, by generating controlled artifacts, it is possible to pave the way for trustworthy AI in industry. This is facilitated by the availability of information and logs for model behavior inspection in case of a system failure. As an additional benefit of this architecture, it is also possible to track the CO_2 emissions of a given DA, which is very relevant for resource intensive solutions employing LLMs. As an example, it can be inspected whether an LLM-based sustainability model emits more CO_2 than it can optimize or not. Nonetheless, since all decisions regarding different DA implementations rely on information from the proposed architecture, automating instance-specific verification remains a challenge for future work. Furthermore, for future work will need to explore LLM-assisted DA creation, integrating operator-defined tasks into an AutoML workflow for task-specific model generation.

7 CONCLUSION

In this paper an architecture for encapsulating information from different DA implementations along with their designated tasks was introduced. We aimed to point out the fact that, the MLOps cycle is a small part of the entire solution development given the overwhelming number of assets and experimental solutions developed for them. Furthermore, we discussed how important it is to prune the artifact generation from different parts of the DA to keep it traceable for debugging as well as data and model aggregation. In addition, we introduced a human-in-the-loop design for AAS creation of a physical asset, aiming for maximized operator involvement in the designing process of different DAs. For future work, we aim to feed prompts containing operator generated tasks with corresponding solutions from the model maintenance submodel to train a LLM for creating new DAs for new tasks. The ultimate goal is to incorporate task descriptions into the AutoML solution development process, making it more relevant to real-world industry use cases.

ACKNOWLEDGMENT

The authors of this paper would like to express their deep sense of gratitude to SMC Schweiz AG, especially Mr. Alessandro Grizzetti, for their unceasing assistance in promoting Competence Centre for Automation and Digitalisation among different industrial partners in Europe. Additionally, we extend our gratitude to the amazing team at ABB OEE team (Marko Barutcu and Patrick Henzi) for their invaluable support in ensuring the success of this project.

REFERENCES

Brettel, M., Klein, M., and Friederichsen, N. (2016). The relevance of manufacturing flexibility in the context of industrie 4.0. *Procedia Cirp*, 41:105–110.

Cerquitelli, T., Nikolakis, N., Morra, L., Bellagarda, A., Orlando, M., Salokangas, R., Saarela, O., Hietala, J., Kaarmila, P., and Macii, E. (2021). Data-driven predictive maintenance: A methodology primer. *Predictive Maintenance in Smart Factories: Architectures, Methodologies, and Use-cases*, pages 39–73.

Colombi, L., Gilli, A., Dahdal, S., Boleac, I., Tortonesi, M., Stefanelli, C., and Vignoli, M. (2024). A machine learning operations platform for streamlined model serving in industry 5.0. In *NOMS 2024-2024 IEEE Network Operations and Management Symposium*, pages 1–6. IEEE.

Dorigo, T., Brown, G. D., Casonato, C., Cerdà, A., Ciarrochi, J., Da Lio, M., D'souza, N., Gauger, N. R., Hayes, S. C., Hofmann, S. G., et al. (2025). Artificial intelligence in science and society: The vision of usern. *IEEe Access*.

Fathi, K., Kleinert, T., and van de Venn, H. W. (2024a). Trustworthy machine learning operations for predictive maintenance solutions. In *PHM Society European Conference*, volume 8, pages 4–4.

Fathi, K., Ristin, M., Sadurski, M., Kleinert, T., and van de Venn, H. W. (2024b). Detection of novel asset failures in predictive maintenance using classifier certainty. In *2024 32nd Mediterranean Conference on Control and Automation (MED)*, pages 50–56. IEEE.

Faubel, L., Schmid, K., and Eichelberger, H. (2023). Mlops challenges in industry 4.0. *SN Computer Science*, 4(6):828.

Ghobakhloo, M. (2020). Industry 4.0, digitization, and opportunities for sustainability. *Journal of cleaner production*, 252:119869.

Haviv, Y. and Gift, N. (2023). *Implementing MLOps in the Enterprise*. " O'Reilly Media, Inc.".

Huang, Y., Dhouib, S., and Malenfant, J. (2021). Aas capability-based operation and engineering of flexible production lines. In *2021 26th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, pages 01–04. IEEE.

ISACA (2024). Understanding the eu ai act: Requirements and next steps. Accessed: 2025-02-06.

Kozma, D., Varga, P., and Larrinaga, F. (2019). Data-driven workflow management by utilising bpmn and cpn in iiot systems with the arrowhead framework. In *2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, pages 385–392. IEEE.

Kreuzberger, D., Kühl, N., and Hirschl, S. (2023). Machine learning operations (mlops): Overview, definition, and architecture. *IEEE access*, 11:31866–31879.

Milicic, A., Kiritsis, D., and Efendioglu, N. (2016). From english to rdf-a meta-modelling approach for predictive maintenance knowledge base design. In *Advances in Production Management Systems. Initiatives for a Sustainable World: IFIP WG 5.7 International Conference, APMS 2016, Iguassu Falls, Brazil, September 3-7, 2016, Revised Selected Papers*, pages 214–224. Springer.

Morgan, J., Halton, M., Qiao, Y., and Breslin, J. G. (2021). Industry 4.0 smart reconfigurable manufacturing machines. *Journal of Manufacturing Systems*, 59:481–506.

Müller, M., Müller, T., Ashtari Talkhestani, B., Marks, P., Jazdi, N., and Weyrich, M. (2021a). Industrial autonomous systems: a survey on definitions, characteristics and abilities. *at-Automatisierungstechnik*, 69(1):3–13.

Müller, T., Jazdi, N., Schmidt, J.-P., and Weyrich, M. (2021b). Cyber-physical production systems: enhancement with a self-organized reconfiguration management. *Procedia CIRP*, 99:549–554.

Müller, T., Lindemann, B., Jung, T., Jazdi, N., and Weyrich, M. (2021c). Enhancing an intelligent digital twin with a self-organized reconfiguration management based on adaptive process models. *Procedia CIRP*, 104:786–791.

Oluyisola, O. E., Sgarbossa, F., and Strandhagen, J. O. (2020). Smart production planning and control: Concept, use-cases and sustainability implications. *Sustainability*, 12(9):3791.

Polke, D., Surjana, A., Diepers, F., Ahle, E., and Söffker, D. (2023). Development of a modular automation framework for data-driven modeling and optimization of coating formulations. In *2023 IEEE 28th International Conference on Emerging Technologies and Factory Automation (ETFA)*, pages 1–8. IEEE.

Psarommatis, F. and May, G. (2024). Digital product passport: A pathway to circularity and sustainability in modern manufacturing. *Sustainability*, 16(1):396.

Raj, E., Buffoni, D., Westerlund, M., and Ahola, K. (2021). Edge mlops: An automation framework for aiot applications. In *2021 IEEE International Conference on Cloud Engineering (IC2E)*, pages 191–200. IEEE.

Ruppert, T., Löcklin, A., Romero, D., and Abonyi, J. (2022). Intelligent collaborative manufacturing space for augmenting human workers in semi-automated manufacturing systems. In *2022 IEEE 27th International Conference on Emerging Technologies and Factory Automation (ETFA)*, pages 1–7. IEEE.

Sabuncu, Ö. and Bilgehan, B. (2025). Human-centric iot-driven digital twins in predictive maintenance for optimizing industry 5.0. *Journal of Metaverse*, 5(1):64–72.

Salama, K., Kazmierczak, J., and Schut, D. (2021). Practitioners guide to mlops: A framework for continuous delivery and automation of machine learning. *Google Could White paper*.

Shakhovska, N. and Campos, J. (2024). Predictive maintenance for wind turbine bearings: An mlops approach with the diafs machine learning model.

Wei, K., Sun, J. Z., and Liu, R. J. (2019). A review of asset administration shell. In *2019 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, pages 1460–1465.

Xia, Y., Shenoy, M., Jazdi, N., and Weyrich, M. (2023). Towards autonomous system: flexible modular production system enhanced with large language model agents. In *2023 IEEE 28th International Conference on Emerging Technologies and Factory Automation (ETFA)*, pages 1–8. IEEE.

Yue, F. and Wang, Y. (2022). Cross-domain fault diagnosis via meta-learning-based domain generalization. In *2022 IEEE 18th International Conference on Automation Science and Engineering (CASE)*, pages 1826–1832. IEEE.

Zhou, B., Nikolov, N., Zheng, Z., Luo, X., Savkovic, O., Roman, D., Soylu, A., and Kharlamov, E. (2023). Scaling data science solutions with semantics and machine learning: Bosch case. In *International Semantic Web Conference*, pages 380–399. Springer.