All You Need is an AI Platform: A Proposal for a Complete Reference Architecture

Benjamin Weigell
University of Augsburg
Augsburg, Germany
benjamin.weigell@uni-a.de

Fabian Stieler
University of Augsburg
Augsburg, Germany
fabian.stieler@uni-a.de

Bernhard Bauer
University of Augsburg
Augsburg, Germany
bernhard.bauer@uni-a.de

***Abstract*—Companies increasingly integrate Artificial Intelligence (AI) into their applications to stay competitive. However, the efficient and successful development and deployment of AI applications requires complex setups. Providing developers with all the required resources, applications, and services for developing and deploying AI applications without reinventing the wheel remains challenging. Therefore, we propose a domain- and workflow-agnostic reference architecture (RA) for an on-premises AI Platform that supports teams throughout the entire AI lifecycle and is reusable across multiple projects. Additionally, we present an evaluation strategy to validate the RA.**

***Index Terms*—AI Platform, MLOps, Software Architecture**

I. Introduction

AI adoption is increasing across industry and academia, especially in the sectors of manufacturing, information, and health care [1]. Concurrent emerging software engineering practices, such as MLOps, have simplified the transition of machine learning (ML) models from development to production [2]. However, following [3], we consider an AI application as a software product that includes ML or other models mimicking learning and problem-solving skills. Consequently, a platform for creating AI applications must address the model lifecycle, as covered by MLOps, and general software engineering needs of building and running applications around those models. This includes transparently managing hardware and infrastructure resources, providing reusable components, and simplifying the implementation and setup of different AI projects through a standardized platform. Additionally, such a platform should avoid domain- or workflow-specific constraints. Various aspects of this have been addressed in the literature. For instance, [4] extended the CRISP-DM model for ML development, and [5] proposed a continuous pipeline for AI model development but focused on the lifecycle and omitted infrastructure management. [6] explored transferring continuous practices to AI/ML development and identified, therefore, necessary components, yet did not propose a standalone platform with infrastructure support and multi-project support. Similarly, [2] presented a MLOps framework outlining core components and an architecture for realizing ML projects; however, it lacked multi-project, infrastructure, and workflow-agnostic functionalities. Above that, existing RAs [7], [8] emphasize automation and governance but fall short in offering multi-project and domain-agnostic project support. Addressing these gaps, we propose an RA of an on-premises AI platform, which we define as: *A Platform-as-a-Service (PaaS) that supports the entire AI lifecycle and multiple AI projects workflow- and domain-agnostically.*

II. Capabilities

We summarize key platform and MLOps capabilities that an on-premises AI RA should support. Six key capabilities have been identified to specify hardware and infrastructure properties required for AI application development, which were derived from the offerings of six cloud providers. The identified capabilities define access to *computation infrastructure* with general computing power and ML accelerators, fast and reliable *storage*, and *networking*. Additionally, they specify *scalable resources* and the provisioning of encapsulated, resource-limited *computation environments*, which can be created *on-demand*. MLOps automates and optimizes the development, deployment, monitoring, and maintenance of ML applications in production [2]. Based on [2] and supported by supplementary literature, we establish four capability groups comprising 13 MLOps capabilities. The Continuous-X group includes Continuous-*Integration* (CI), -*Delivery* (CD), -*Deployment* (CDP), -*Training* (CT), and -*Monitoring* (CM). CI integrates code changes with automated code testing, data validation, and model convergence checks. CD and CDP ensure deployment readiness with additional quality checks and seamless integration of trained models into software systems [9]. CT re-executes ML pipelines based on triggers from the CM system, monitoring system performance, data quality, and model behavior [2], [5], [9]. The next group, Cross-Functional Development, is supported by *Collaboration*, promoting cooperation across code, data, and models [2], and *User Friendliness*, ensuring accessibility to all stakeholders. The Orchestration and Automation group includes *Orchestration*, structuring and executing ML pipeline stages [2], and *Automation*, enhancing workflow automation from development to model integration [6]. Finally, the Traceability and Reproducibility group ensures traceability through *Versioning* of code, data, and models [5] and *Metadata Capture* for models, pipelines, and overall dependencies [2], [5], [6]. This

This work was partially funded by the German Federal Ministry of Education and Research (BMBF) under reference number 031L9196B. Transparency Note: The readability of this text has been improved with the support of AI technologies.

traceability enables *Reproducibility*, ensuring results can be replicated across different environments [9].

III. AI Platform Reference Architecture

We propose an RA for an AI Platform structured into three layers as shown in Figure 1: The **Hardware Layer** provides the physical resources, including CPU, RAM, ML accelerators, storage, and network devices, to the higher layers. The **Infrastructure Layer** connects the hardware and platform layers by pooling physical resources and abstracting their usage. These resources are aggregated into a Cluster, and Virtualization is used for resource isolation and multi-tenancy of computation environments (CEs). A Resource Scheduler optimizes CE placement, and a Cluster State Watcher ensures high availability. Additionally, Storage, Accelerator, and Network Managers manage access to their respective hardware functionalities. Finally, a Management Endpoint simplifies deployment, management, and monitoring. The **Platform Layer** forms the core of the AI Platform, designed to support teams throughout the entire AI lifecycle workflow- and domain-agnostically in various projects. To achieve this, it offers 14 components organized into five categories. The purpose of the components in the *Artifact Management* group - Source Code Management (SCM), Data Store, Metadata Store, Model Registry, and Image Registry - is to manage artifacts while ensuring complete model lineage. Next, in the *Automation* group, CI/CD components automate testing and deployment, while the CM system monitors deployed models and triggers the CT system for retraining. A Workflow Orchestrator optimizes ML pipeline execution, and an Image Builder automates the creation of images for CEs. The *Identity Management* group includes an Identity Provider, which facilitates authentication and resource sharing. The *Project Spaces* group provides a Zone Manager that creates and manages isolated, resource-restricted zones that encapsulate multiple CEs. These zones provide dedicated environments for projects, development, and deployment workflows. In the group, *CE Provisioning*, a Local Client coordinates user interaction with the platform to provide CEs connected to the user's IDE.

IV. Planned Evaluation

We plan to evaluate the RA across three key aspects with two theoretical scenarios and one practical instantiation. In the first scenario, we will assess the ability of the RA to support AI/ML development workflows such as CRISP-ML(Q) [4]. With the second scenario, we will determine whether the RA supports the seven principles of trustworthy AI development outlined in [10]. Finally, to examine reusability and domain-agnostic properties, an instance of the RA and two AI applications, one for image generation and one for signal data analysis, will be developed. Insights from this instance evaluation will help determine if the RA supports reusability, domain-agnostic development, and practical instantiation.

V. Discussion and Conclusion

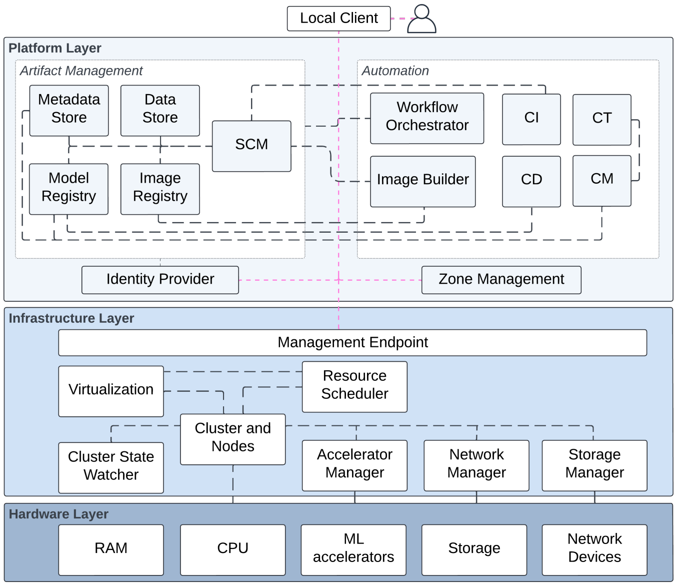

Fig. 1. Reference Architecture of the AI Platform

The proposed RA for an on-premises AI PaaS is built upon general platform and MLOps-specific capabilities. It employs a three-layer structure where the infrastructure layer transparently connects the hardware and platform layers. The architecture is designed to impose no domain or workflow constraints, which will be validated through the proposed evaluation. Additionally, the emphasis on IAM, virtualization, and independent zones further promotes the reuse of resources across multiple projects. However, relying solely on general platform capabilities from cloud providers could introduce bias. The planned RA instantiation will assess this dependency.

References

[1] K. McElheran, J. F. Li, E. Brynjolfsson, Z. Kroff, E. Dinlersoz, L. Foster, and N. Zolas, "AI adoption in America: Who, what, and where," *Journal of Economics & Management Strategy*, vol. 33, no. 2, 2024.

[2] D. Kreuzberger, N. Kühl, and S. Hirschl, "Machine Learning Operations (MLOps): Overview, Definition, and Architecture," *IEEE Access*, vol. 11, pp. 31 866–31 879, 2023.

[3] I. Ozkaya, "What is really different in engineering ai-enabled systems?" *IEEE Software*, vol. 37, no. 4, p. 3–6, 2020.

[4] S. Studer, T. B. Bui, C. Drescher, A. Hanuschkin, L. Winkler, S. Peters, and K.-R. Müller, "Towards CRISP-ML(Q): A Machine Learning Process Model with Quality Assurance Methodology," *Machine Learning and Knowledge Extraction*, vol. 3, no. 2, pp. 392–413, 2021.

[5] M. Steidl, M. Felderer, and R. Ramler, "The pipeline for the continuous development of artificial intelligence models—Current state of research and practice," *Journal of Systems and Software*, vol. 199, 2023.

[6] L. E. Lwakatare, I. Crnkovic, E. Rånge, and J. Bosch, "From a Data Science Driven Process to a Continuous Delivery Process for Machine Learning Systems," pp. 185–201, 2020.

[7] I. Kumara, R. Arts, D. D. Nucci, W. J. V. D. Heuvel, and D. A. Tamburri, "Requirements and reference architecture for mlops:insights from industry," *Authorea Preprints*, 2023.

[8] T. Raffin, T. Reichenstein, J. Werner, A. Kühl, and J. Franke, "A reference architecture for the operationalization of machine learning models in manufacturing," *Procedia CIRP*, vol. 115, p. 130–135, 2022.

[9] M. Testi, M. Ballabio, E. Frontoni, G. Iannello, S. Moccia, P. Soda, and G. Vessio, "MLOps: A Taxonomy and a Methodology," *IEEE Access*, vol. 10, pp. 63 606–63 618, 2022.

[10] E. Commission, D.-G. for Communications Networks, Content, and Technology, *Ethics guidelines for trustworthy AI*. Publications Office.