*Article*

# AI Lifecycle Zero-Touch Orchestration within the Edge-to-Cloud Continuum for Industry 5.0

Enrico Alberti [1,†] , Sergio Alvarez-Napagao [2,†] , Victor Anaya [3,†] , Marta Barroso [2,†] , Cristian Barrué [4,†] , Christian Beecks [5,†] , Letizia Bergamasco [6,†] , Sisay Adugna Chala [7,†] , Victor Gimenez-Abalos [2,†] , Alexander Graß [7,†] , Daniel Hinjos [2,†] , Maike Holtkemper [5,†] , Natalia Jakubiak [4,†] , Alexandros Nizamis [8,†] , Edoardo Pristeri [6,†] , Miquel Sànchez-Marrè [4,†] , Georg Schlake [5,†] , Jona Scholz [5,†] , Gabriele Scivoletto [1,†] and Stefan Walter [9,*,†]

1   Nextworks Srl, Via Livornese 1027, 56122 Pisa, Italy; e.alberti@nextworks.it (E.A.); g.scivoletto@nextworks.it (G.S.)
2   Barcelona Supercomputing Center, Plaça Eusebi Güell 1-3, 08034 Barcelona, Spain; sergio.alvarez@bsc.es (S.A.-N.); marta.barroso@bsc.es (M.B.); victor.gimenez@bsc.es (V.G.-A.); daniel.hinjos@bsc.es (D.H.)
3   Information Catalyst SL, Cl Reina 27, 4-7, 46800 Xativa, Spain; victor.anaya@informationcatalyst.com
4   Department of Computer Science, IDEAI Research Centre, Universitat Politècnica de Catalunya (UPC), Carrer Jordi Girona 1-3, 08034 Barcelona, Spain; cbarrue@cs.upc.edu (C.B.); jakubiak@cs.upc.edu (N.J.); miquel@cs.upc.edu (M.S.-M.)
5   Department of Data Science, University of Hagen, 58097 Hagen, Germany; christian.beecks@fernuni-hagen.de (C.B.); maike.holtkemper@fernuni-hagen.de (M.H.); georg.schlake@fernuni-hagen.de (G.S.); jona.scholz@fernuni-hagen.de (J.S.)
6   LINKS Foundation, Via Pier Carlo Boggio 61, 10138 Torino, Italy; letizia.bergamasco@linksfoundation.com (L.B.); edoardo.pristeri@linksfoundation.com (E.P.)
7   Fraunhofer Institute for Applied Information Technology (FIT), Schloss Birlinghoven, 53757 Sankt Augustin, Germany
8   Centre for Research and Technology Hellas-Information Technologies Institute (CERTH/ITI), Charilaou-Thermis, 57001 Thessaloniki, Greece; alnizami@iti.com
9   VTT Technical Research Centre of Finland Ltd., Tekniikantie 21, 02150 Espoo, Finland
*   Correspondence: stefan.walter@vtt.fi
†   All authors contributed equally to this work.

**Abstract:** The advancements in human-centered artificial intelligence (HCAI) systems for Industry 5.0 is a new phase of industrialization that places the worker at the center of the production process and uses new technologies to increase prosperity beyond jobs and growth. HCAI presents new objectives that were unreachable by either humans or machines alone, but this also comes with a new set of challenges. Our proposed method accomplishes this through the knowlEdge architecture, which enables human operators to implement AI solutions using a zero-touch framework. It relies on containerized AI model training and execution, supported by a robust data pipeline and rounded off with human feedback and evaluation interfaces. The result is a platform built from a number of components, spanning all major areas of the AI lifecycle. We outline both the architectural concepts and implementation guidelines and explain how they advance HCAI systems and Industry 5.0. In this article, we address the problems we encountered while implementing the ideas within the edge-to-cloud continuum. Further improvements to our approach may enhance the use of AI in Industry 5.0 and strengthen trust in AI systems.

**Keywords:** Industry 4.0; human-centered AI; manufacturing; model orchestration; digital twin

## 1. Introduction

Modern manufacturing faces a complex array of challenges that impact efficiency, sustainability, and competitiveness [1–5]. Supply chain disruptions, heightened by the recent pandemic, have highlighted the need for greater resilience and flexibility [6–9]. The

reliance on global networks, coupled with uncertainties in transportation and demand, underscores the importance of adaptive strategies and real-time data analytics [10–12].

In the ever-evolving landscape of manufacturing, a multitude of interconnected challenges shape the industry's ability to meet the demands of today's dynamic markets. These challenges encompass diverse facets, each playing a pivotal role in the pursuit of operational excellence and sustainable growth [4,13,14]. In a comprehensive analysis, the preceding challenges could be categorized as encompassing product quality, process quality, process planning and scheduling, adaptability within the context of market dynamics, and interaction between humans and machines, as well as the quality and accessibility of data [11,15–18].

At the heart of manufacturing lies the paramount concern of product quality. Delivering products that consistently meet or exceed customer expectations remains a fundamental objective. Achieving this requires stringent quality control measures, efficient defect detection systems, and a proactive approach to identifying and rectifying issues. However, maintaining product quality becomes even more complex as manufacturing processes become increasingly intricate and technologically advanced [19,20].

Process quality, closely intertwined with product quality, stands as a cornerstone of effective manufacturing. Ensuring that each step of the production process adheres to established standards is crucial. The challenge here often lies in implementing robust quality management systems that monitor, analyze, and optimize processes in real time [21].

In the context of process planning and scheduling, striking the delicate balance between maximizing resource utilization and meeting delivery timelines is a perpetual challenge. Optimizing process planning and scheduling involves not only efficient resource allocation but also the ability to adapt with agility to unforeseen disruptions. As markets fluctuate and demand patterns shift, the manufacturing industry grapples with the need for agile planning systems that can accommodate rapid changes without sacrificing efficiency [22–24].

Flexibility within manufacturing industries is paramount in navigating the turbulent waters of a rapidly changing market. The ability to swiftly pivot production lines, reconfigure processes, and adjust product offerings in response to market shifts is a formidable task [4,25,26].

In addition, as automation and robotics become more prevalent, ensuring harmonious collaboration between human workers and machines is essential. Designing intuitive and user-friendly interfaces, training workers to interact seamlessly with automated systems, and fostering a culture that embraces technological integration while valuing human expertise is key to maximizing the benefits of HMI [27–29].

In this digital era, the availability and quality of data form the bedrock of effective decision-making. However, the manufacturing industry often faces hurdles related to data quality and accessibility. Overcoming data silos, ensuring data integrity, and establishing robust data collection and management practices are essential to unlock the full potential of data-driven insights [30,31].

This paper endorses the relevance that cloud-edge computing has as a key enabler for Industry 5.0, allowing for real-time data processing and analysis at the edge of the network (closer to where the data is generated) while intensive computation tasks are carried out in the cloud. This is where AI lifecycle zero-touch orchestration comes into play, providing a framework for managing the deployment, scaling, and maintenance of AI applications across the edge-to-cloud continuum [32] and how it can be used to enable intelligent, efficient, and reliable solutions in smart manufacturing. The article also examines the role of embedded artificial intelligence in edge computing and how it can be used to optimize resource management and innovate production processes [33].

The main contribution of our paper is a novel approach to AI lifecycle orchestration that emphasizes human-AI collaboration. Our approach is extended through a digital framework, which allows domain experts to train, test, and evaluate AI models even before bringing them into production. Furthermore, we developed an AI model recommendation

system that uses semantic reasoning to provide more understandability and controllability to the human expert.

This article is structured as follows: Section 2 is where related work is introduced to establish the existing landscape. Sections 3 and 4 offer a theoretical understanding and practical implementation details of the AI lifecycle within the knowlEdge framework. Following this, Section 5 illustrates specific use cases and their applications across various industries. Section 6 delves into the intricacies of the design and deployment of the knowlEdge project. The subsequent section, Section 7, engages in a thorough discussion, describing the significance of the knowlEdge findings and contributions in relation to human-centered artificial intelligence. Finally, Section 8 serves as a conclusion, summarizing the key findings and proposing avenues for further exploration in the realm of AI for manufacturing.

## 2. Related Works

The state of the art in digital industrial platforms and smart service systems has been extensively explored in recent research. AI model lifecycle management has become increasingly important in the development of Industry 5.0. The management of AI models throughout their lifecycle is critical to ensure their effectiveness and efficiency [34]. The integration of zero-touch orchestration frameworks into the edge-to-cloud continuum has enabled the development of AI model lifecycle management systems that can autonomously manage and orchestrate AI models throughout their lifecycle [35], i.e., data acquisition, model development, human collaboration, and model sharing.

Ref. [36] addresses digital industrial platforms and highlights their value in promoting innovation and teamwork within industrial ecosystems. In an investigation into the growth of industrial IoT platforms, Ref. [37] discusses the trade-offs between horizontal and vertical approaches. The authors stress the significance of striking a balance between general functionality and particular industrial requirements. Ref. [38] focuses on the challenges, opportunities, and directions in the field of the Industrial Internet of Things (IIoT), such as interoperability, security, data analytics, and standardization. Collectively, these studies contribute to a comprehensive understanding of the advancements and complexities in digital industrial platforms and smart service systems, providing valuable insights for researchers and practitioners in the field.

The ubiquity that the Internet of Things (IoT) has reached in recent years has drawn attention to the execution of AI models, even at the edge of the networks. Many works focus on the continuous training and deployment of AI models across edge, fog, and cloud systems to leverage the specific advantages of all the different computing environments. In [39], the authors present an automated framework for machine learning operations in edge systems called Edge MLOps, which combines cloud and edge environments, enabling continuous model training, deployment, delivery, and monitoring. The framework is validated in a real-life scenario for the forecast of room air quality. Another relevant framework for the automation of AI learning processes and deployment management is the complex event machine learning framework (CEML) [40], developed to be suitable for deployment in any environment, from the edge to the cloud. In particular, CEML allows for the continuous execution of all the phases of the AI learning process: data collection from different sensors, data preprocessing and feature extraction, learning, learning evaluation, and deployment (this takes place when the model is ready, according to the learning evaluation).

Inducing domain knowledge into AI models is important to produce a more complete solution by augmenting the learned knowledge acquired by the models. It is used to improve the accuracy of predictive models, e.g., to guide feature selection in a machine learning model, resulting in better predictive performance [41] and improved effectiveness regarding predictive models.

One aspect of human feedback to an AI model is fact-checking. Fact-checking, a task to evaluate the accuracy of AI models, is a crucial, pressing, and difficult task. Despite the

emergence of numerous automated fact-checking solutions, the human aspect of this collaboration has received minimal attention [42], though some advancement is being observed in conversational AI [43]. Specifically, it remains unclear how end users can comprehend, engage with, and build confidence in AI-powered agile manufacturing systems.

Existing studies on human-AI collaboration predominantly focus on UI and UX aspects, i.e., whether (and how) the AI systems provide an intuitive user interface. A number of them assessed human-AI collaboration with respect to human-AI interaction guidelines as opposed to features that enable a human actor to provide feedback to the AI model [44,45]. The effect of users' decision-making has been has been studied using different explainable AI (XAI) interface designs [46].

There are multiple other projects that offer an AI marketplace, such as NVIDIA Triton[1], the Verta Model registry[2], modelplace.ai[3], the Modzy AI Model Marketplay[4], and SingularityNET[5]. However, these marketplaces are purely based on the idea of sharing existing models. Our approach of linking datasets and models to create a valuable storage of information for the development of new models is not present in either of these projects.

The knowlEdge framework adds to the Gaia-X[6] and data space initiatives[7] by demonstrating the AI lifecycle for industrial and manufacturing use cases. The framework of knowlEdge follows the decentralized and open approach of GAIA-X in that the services are also offered for on-demand execution and orchestration, and the storage of models and data in the nodes of a cloud are of central importance. More specifically, knowlEdge's AI lifecycle and GAIA-X are related in the context of how AI solutions are developed, deployed, and managed within a framework that emphasizes data sovereignty, fostering interoperability, collaboration, and trust. As such, the knowlEdge framework implements components that ensure data sovereignty and orchestration (collecting data from data sources, modeling training environments, and deploying platforms). With respect to trust, knowlEdge implements transparent processes and model explainability. Finally, knowlEdge's AI lifecycle also implements a human-in-the-loop approach, which facilitates collaboration between different stakeholders, including data scientists, domain experts, and IT professionals.

## 3. AI Lifecycle: A Theoretical Exploration

The AI lifecycle refers to the entire process of developing and deploying AI solutions, starting from problem identification to the eventual deployment of the solution and its maintenance. It encompasses all the stages involved in creating an AI system, including data acquisition, model development, evaluation, deployment, ongoing monitoring, and maintenance. Machine learning is a subset of AI, which focuses on algorithms that learn from data. Note that the terminology in this paper may alter between the two, depending on whether we are referring to the broader concept or the narrower subset. Furthermore, we make an attempt to reflect the conventions of different application contexts, which may vary in their use of each term.

The increasing adoption of machine learning in real-world applications has created new ways of deploying and managing machine learning models in production environments. MLDevOps emerges as an approach that addresses the unique challenges that arise when deploying and managing machine learning models while ensuring scalability, reliability, and maintainability.

Within this landscape, the edge-to-cloud AI lifecycle represents a comprehensive framework that spans the entire journey of developing, deploying, and refining AI solutions that span from centralized data centers (the cloud) to the very edge of a network. It aims to optimize AI applications for diverse environments, considering factors such as latency, bandwidth, and privacy, thereby providing a holistic and adaptive approach to AI development and deployment. This approach stands in contrast to the more common practice where AI applications are developed and run entirely in the cloud.

Typically, the AI lifecycle process consists of the following steps:

- Problem definition: Clearly defining the problem or objective that the AI system aims to solve or achieve. This step involves understanding the requirements, constraints, and desired outcomes.
- Data acquisition: Gathering the relevant data necessary to train and build the AI models.
- Data preparation: Preprocessing and cleaning the acquired data to ensure they are in a suitable format for analysis. This step may involve tasks such as data cleaning, data integration, data transformation, and handling missing values.
- Model development: Creating and training the AI model using the prepared data. This step involves selecting an appropriate algorithm or model architecture, splitting the data into training and validation sets, and iteratively training and refining the model.
- Model evaluation: Assessing the performance and effectiveness of the trained model. This step involves evaluating the model's accuracy, precision, recall, and other relevant metrics using appropriate evaluation techniques. It helps to determine whether the model meets the desired performance criteria.
- Model deployment: Integrating the trained model into a production environment where it can be used to make predictions or provide insights. This step involves deploying the model on appropriate infrastructure, setting up necessary interfaces, and ensuring scalability, reliability, and security.
- Monitoring, maintenance, and distribution: Continuously monitor the deployed model's performance in real-world scenarios and make necessary adjustments or updates as needed. This step involves tracking key performance metrics, handling model drift, addressing issues, and periodically retraining or fine-tuning the model. At this stage, models can also be distributed to other individuals or organizations. Model sharing can be facilitated through various means, such as publishing models in open source repositories, sharing code repositories, using model exchange formats like ONNX (open neural network exchange), or providing APIs and web services for accessing and utilizing the models. It may involve sharing the trained model parameters, architecture, or even the entire model pipeline with others who may benefit from using or further developing the model.

The upcoming section presents how each step in the AI lifecycle is implemented in the knowlEdge project framework, explaining the specific roles and functions of the components involved. From collecting and preparing data to deploying and maintaining models, each phase provides a clear picture of how AI technologies work across different applications.

## 4. AI Lifecycle Implementation: The knowlEdge Framework

In the context of the AI lifecycle , the knowlEdge project, cf. [47], H2020 Framework Programme of the European Commission, under grant agreement 957331, proposes a framework that allows for the development of applications empowered with AI techniques and technologies that take advantage of the compute continuum to support distributed heterogeneous scenarios. The framework is based on components at different functional levels, covering technologies such as manufacturing digital twins, smart decision-making dashboards, advanced AI algorithm recommenders based on historical performance, or human-AI collaboration and domain knowledge fusion.

The following subsections explain the main functionalities provided by the knowlEdge's components that are more concerned with the AI lifecycle described before. These components are highlighted in blue in Figure 1.

### 4.1. Data Acquisition and Preparation Data Pipeline

The AI lifecycle starts with the execution of a data pipeline, which encompasses data collection, data distribution, and data storage, guaranteeing the systematic and secure aggregation of data from sensors. This process ensures the authenticity of the gathered data, thereby facilitating precise analysis and informed decision-making. Subsequently, data distribution ensures the secure and efficient dissemination of data across distinct

sections of the network (edge, fog, and cloud). Through the implementation of robust security protocols, data can be transmitted securely, preserving privacy and maintaining confidentiality. Finally, data storage assures the secure preservation of data over extended periods. By employing reliable storage mechanisms, valuable information is safeguarded, enabling retrospective analysis and facilitating historical comparisons. Collectively, the integration of these three functions within a data pipeline ensures the trustworthiness, security, and accessibility of data used for training and those used when running AI models, thereby enhancing their efficiency and effectiveness. Colours of the different layers at Figure 1 are not meaningful, and it is only a way to differentiate among them.
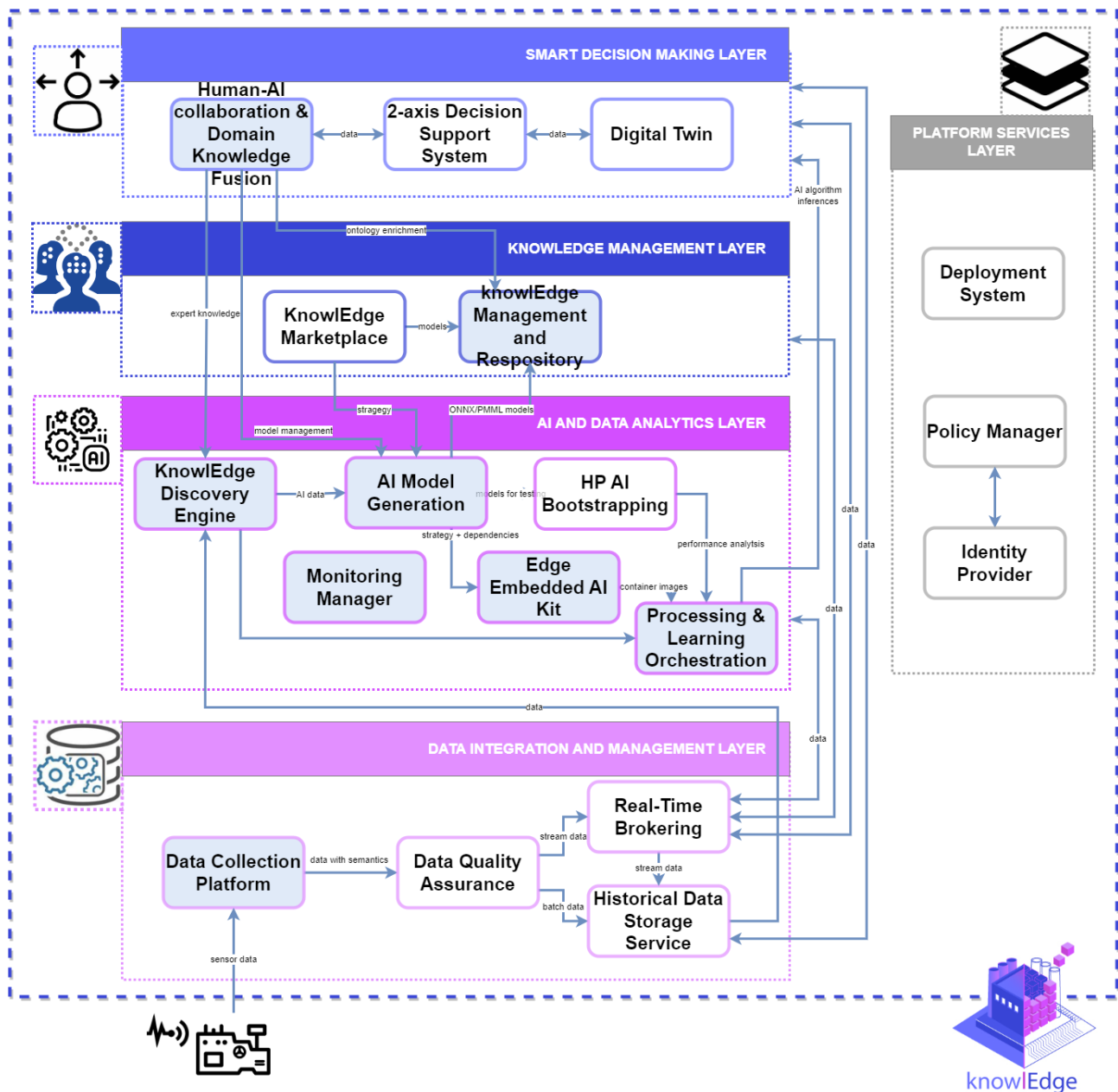


**Figure 1.** An overview of the knowlEdge architecture courtesy of [48].

### 4.1.1. Data Collection

The Data Collection Platform, a key element of the knowlEdge architecture, facilitates the gathering of diverse data from the various shop floors within the knowlEdge pilots. Its purpose is to translate information obtained from different factory systems into a standardized data model, ensuring seamless communication with knowlEdge applications.

The collected data is subsequently distributed in real-time to the higher-level components of the knowlEdge system, serving as data consumers.

The Data Collection Platform was initially constructed using Symphony[8], a commercial IoT platform and building management system (BMS) developed by Nextworks. Symphony boasts a modular architecture, making it a comprehensive IoT platform and BMS that facilitates seamless and unified interaction with a diverse range of hardware devices, IoT sensors, and actuators. The Hardware Abstraction Layer (the architecture of which is explained in Figure 2, where boxes in blue represent Data Platform components, white boxes in grey represent external components) a key component of Symphony, serves as the data collection and processing module by abstracting low-level connectivity details, enabling seamless data collection while interacting with a wide array of hardware devices, IoT sensors, and actuators.

In an Internet of Things (IoT) system, the components work together to establish seamless connectivity and functionality ([49]). These components consist of sensors, actuators, devices, gateways, and the IoT integration middleware.
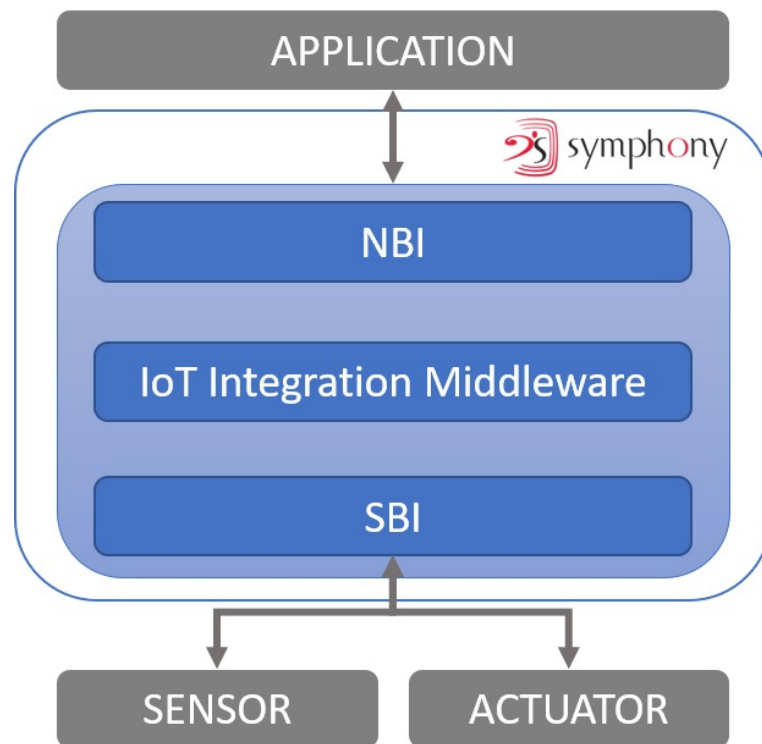


**Figure 2.** The Symphony IoT platform high-level architecture.

Sensors are hardware components that measure parameters in the physical environment and convert them into electrical signals, capturing data such as temperature or humidity. Actuators, on the other hand, actively control or manipulate the physical environment by emitting optic or acoustic signals. When devices face technical limitations for direct connections, gateways serve as intermediaries, providing the necessary technologies and functionalities to overcome these limitations. Gateways enable bidirectional communication, translating data between devices and systems and resolving protocol incompatibilities. The IoT integration middleware plays a vital role in the IoT system by receiving and processing data from connected devices, including evaluating condition-action rules. It facilitates the provision of processed data to applications and exercises control over devices through commands sent to the actuators. Direct communication with the middleware is possible for compatible devices, while devices lacking compatibility communicate through gateways. Thus, the IoT integration middleware acts as an integration layer, harmonizing

interactions between sensors, actuators, devices, and applications in the IoT ecosystem, ensuring seamless connectivity and integration.

Symphony, as an implementation, plays a crucial role in this system by providing a driver with a southbound interface (SBI) and a gateway with a northbound interface (NBI). The driver, equipped with the southbound interface, enables communication between Symphony and the lower-level components, such as sensors and actuators, facilitating data collection and control. On the other hand, the gateway, featuring the northbound interface, establishes connectivity with higher-level devices or networks, allowing Symphony to exchange information and commands seamlessly. By integrating these interfaces, Symphony ensures efficient communication between the diverse IoT components, enabling co-ordinated operations within the IoT system.

Both the southbound and northbound plugins incorporate Integration Middleware, which preprocesses and harmonizes the data samples based on the knowlEdge data model.

Its flexibility and modularity were improved through internal enhancements. Furthermore, new southbound plugins and northbound interfaces (NBIs) were added to enable seamless interaction with other components within the knowlEdge architecture.

### 4.1.2. Data Management and Distribution

After the data are collected, they first pass through the Data Quality Assurance component, as illustrated in Figure 1. This edge-deployed component is crucial for performing preprocessing and anonymization of the data before storage and distribution outside of the secure industrial network. The primary purpose of this component is to ensure compliance with appropriate levels of data quality and to safeguard information while maintaining the usability and accessibility of the data. The Data Quality Assurance component supports both suppression and pseudonymization methods. Suppression involves the removal of features that are not critical for machine learning/deep learning tasks and could potentially lead to the re-identification of the data subject. On the other hand, pseudonymization is a process where one attribute in a record, typically a unique one, is replaced with another. This is achieved through techniques such as secret key encryption, hash functions, and tokenization. These methods play a vital role in preventing the identification of personally identifiable information (PII) and other sensitive production data, thus ensuring the privacy and non-identifiability of data subjects.

Following this quality assurance, the data are distributed via a secure message broker supported by MQTTs (MQTT over SSL/TLS), and they are saved into a distributed data store that spans both edge and cloud environments. Acting as an intermediary, the message broker efficiently routes the data to their intended destinations, utilizing its queuing mechanisms and flexible routing capabilities while satisfying security levels that are adequate for industrial applications. This is shown in Figure 3. The distributed data store encompasses both edge devices and cloud infrastructure, enabling organizations to leverage the advantages of edge computing, where data processing occurs closer to its source, reducing latency and enhancing real-time decision-making capabilities. Additionally, cloud storage provides scalability, resilience, and centralized management. By combining edge and cloud storage, this approach establishes a robust and adaptable architecture capable of handling large data volumes while ensuring optimal performance and accessibility.

Data availability is contingent upon several factors within the network, including the segment it resides in (cloud, fog, or edge), the access level of the software module, and the specific needs of the consumer. For instance, historical data are essential for AI/ML training and data visualization purposes, whereas live data is crucial for AI/ML inference and prediction tasks. Moreover, the level of data protection required also influences its availability. As explained at the beginning of this subsection, to adhere to privacy and confidentiality standards, the data samples undergo preprocessing and anonymization before storage and distribution. This ensures compliance with the appropriate levels of privacy and confidentiality, safeguarding sensitive information while maintaining the usability and accessibility of the data.
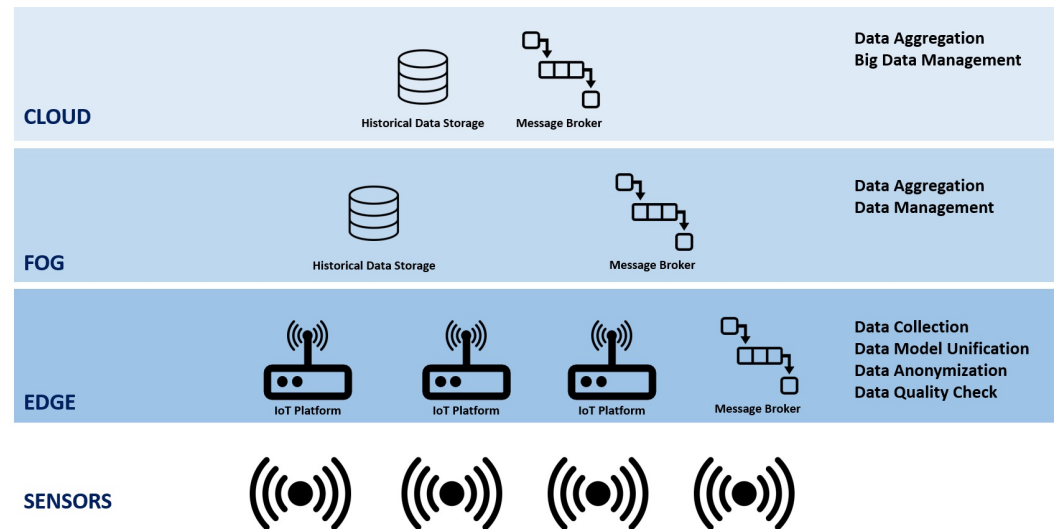
**Figure 3.** Data Management on the edge/fog/cloud.

*4.2. AI Model Development, Evaluation, and Deployment*

This section covers the knowlEdge components in charge of the training, evaluation, deployment, and monitoring of AI models. More precisely, it can be divided into the following components:

- Knowledge Discovery: This component is devoted to data exploration and knowledge discovery, which also includes feature extraction, anomaly or motif detection, and time series structure discovery.
- AutoML Algorithm Selection/Recommendation: When given a problem to solve (task), this component selects the most appropriate algorithm by taking into account the type of a problem (classification, regression, or optimization) and some performance metrics to be maximized. Once the expert has set the final list of algorithms, they can send a request for training to the AI Model Generation component.
- AI Model Generation: This component generates AI models to solve user-defined tasks based on an initial configuration in which the data source, the type of problem, and the algorithm are specified. In addition, it enables the computation of the algorithms' training costs using a set of heuristics, which serves as input to the previous component to provide the recommendations.
- Model Orchestration: This component is responsible for handling the overall deployment and monitoring of AI models, including orchestration, execution, and evaluation.

4.2.1. knowlEdge Discovery

The knowlEdge Discovery Engine (KDE) provides different methods for data exploration based on data from heterogeneous sources. As its core functionality involves the identification of explicit and latent data characteristics, the KDE primarily uses approaches from the fields of unsupervised data analysis. A full overview is given in Figure 4.

One example of the aforementioned data characteristics is the occurrence of anomalies in time series data. The KDE can be trained in an unsupervised way to learn the expected characteristics of the data and predict whether a given data point is anomalous or not. To that end, users can choose between several available models and configure the task to their needs.

All tasks are accessible through one unified API, where a configuration determines which task is executed and the corresponding set of associated parameters. An example of such a configuration is shown in Figure 5. In addition, the configuration allows the user to compose subtasks in a customized manner by combining them in a pipeline of individual subtasks.

The KDE supports a variety of connectors for different data sources. Most notably, the user can load data from the Data Collection Platform (see Section 4.1.1), a broker, or even from a file. New integrations can easily be added, as a data source is just a modular component with a set of parameters.

The API is implemented through a Flask [50] web service. Upon receiving a request, the configuration is parsed and the task is started. Since it is preferable to run tasks in isolation, we employ Docker [51]. Starting a task involves creating a container from a task image, where a container receives the task configuration and builds all the required components to perform its analysis. In order to monitor the status of the process, a task image not only facilitates the execution of a desired analysis specified by a configuration, but also incorporates a web service for each task. These web servers offer multiple endpoints that allow for checking the progress of execution via an API.
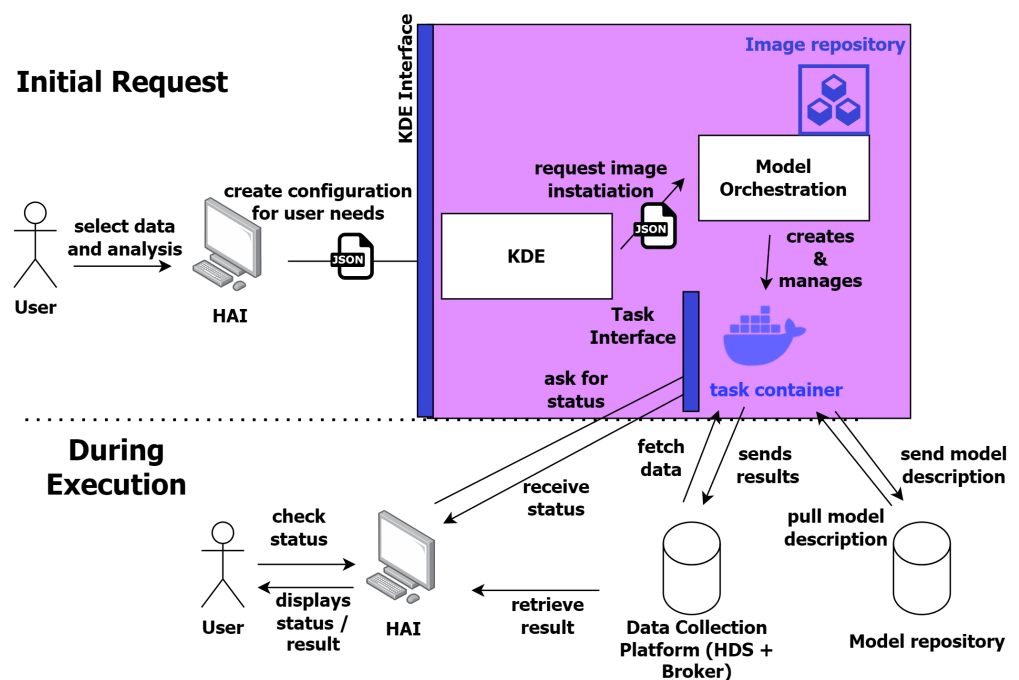


**Figure 4.** A high-level overview of the knowlEdge discovery engine. Users interact with the KDE through the HAI component. This component converts the user inputs into a JSON configuration and sends it to the KDE. The KDE, in turn, builds a task image from the configuration through the Model Orchestration component.

Different actions, such as training or inference processes, can be executed in parallel and will henceforth be referred to as subtasks. This allows for the preprocessing of data and more complex analysis. The user can specify them in the configuration file as a list of ordered objects. Each subtask has an identifier, an action to perform, and the name of a data source. There are two important design decisions related to this setup. First, a subtask does not have to terminate to allow for continuous incremental training. Second, subtasks can have the same order to enable parallelism. Consequently, one can specify a pipeline that performs training on historical data (order 1), trains a model from live data (order 2), and, at the same time, allows for continuous predictions whenever new data are available (also order 2). The latter two subtasks will never terminate, allowing the system to make predictions on new data indefinitely.

The models are wrapped in the convention of Sklearn [52] objects to enable interoperability. They can be saved in the PMML[9] format and, thus, allow for direct integration with the knowlEdge Repository (see Section 4.4).

4.2.2. AI Model Generation

The AI Model Generation (AMG) component is responsible for the automatic creation of supervised AI models that can solve tasks based on various scenarios and input variables. The model development process involves the execution of a whole chain of subprocesses, including data loading from the Data Collection Platform (see Section 4.1.1) or a data stream, automated data preprocessing (autoML), cost computation, automatic model hyperparameter tuning (autoML), training, inference, explainability generation, standardization, and containerization. Each of these stages constitutes a submodule by itself. In the same way as the KDE (see Section 4.2.1), as input, the component receives a configuration JSON file, including information about the algorithms to be trained, the type of problem to be solved, the data source, the type of validation, and the performance metrics, among others. Once the models have been containerized, they can be deployed effectively in both high-performance computing (HPC) and cloud environments and are stored in the knowlEdge Repository (see Section 4.4) for future use.

```
{
''task'': ...,
''method'': ...,
''processing'': [
{
''order'': 1,
''action'': ''train'',
''read'': #hist.-data
},
{
''order'': 2,
''action'': ''train'',
''read'': #live-data
},
{
''order'': 2,
''action'': ''predict'',
''read'': #live-data,
''write'': #result
}
]
}
```

**Figure 5.** An example of a KDE configuration. Task and method are redacted, and the read and write fields are filled with placeholders.

The overall logic of AMG, depicted in Figure 6, where boxes in grey represent AI Model Generation components, and purple boxes represent other knowlEdge components, can be broken down into the various steps that make up the AI lifecycle: the Automatic Preprocessing module, Cost Computation module (estimation of training cost for a specific algorithm), Automatic Hyperparameter Tunning module, Automatic Training, Inference, and Standardization, Explainability module (generation of local and global explanations), Pipeline Execution module (constitutes the main program that calls the rest of modules), and Edge Embedded AI Kit (builds Docker images for model deployment).

In the first stage, the preprocessing module transforms raw data into a format that is suitable for analysis and machine learning. This includes tasks such as normalization, scaling, feature extraction, and selection. The goal of data preprocessing is to improve the quality and relevance of the data, reduce noise, and remove inconsistencies or errors that may negatively impact the accuracy and performance of machine learning models. Regarding data types, the supported formats include tabular, time series, and image data.

In order to give some insights into the training complexity of the algorithms prior to training, the Cost Computation module is able to compute the training cost based on the history of the previously trained models. Execution traces have been generated to describe the behavior and parallelism, as well as the resource usage of an initial set of models. Traces are used mainly to analyze executions but also to provide insights in order to modify model implementation or execution and environment configurations. Afterward, this information is fed to the AutoML Algorithm (see Section 4.2.3) to rank algorithm recommendations.
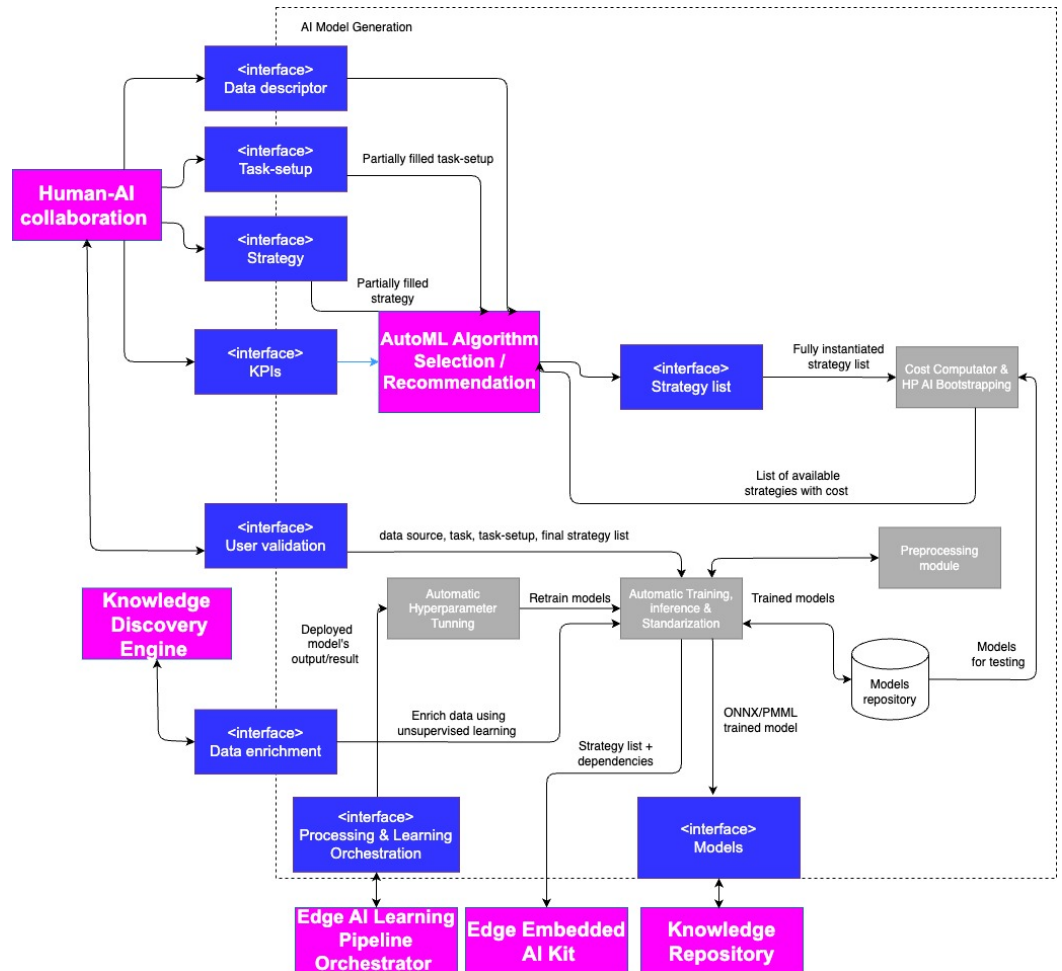


**Figure 6.** AI model low-level architecture.

Once the data have been preprocessed and the algorithm has been selected (through the configuration file or via the recommender (see Section 4.2.3)), the next step is to choose the set of hyperparameters to execute the training. For that purpose, the Automatic Hyperparameter Tuning module provides automated methods for finding the best combination of hyperparameters that optimize the performance of machine learning and deep learning models devoted to classification, regression, and optimization tasks. Hyperparameters are configuration settings that cannot be learned directly from the training data, such as the learning rate, regularization strength, or number of hidden layers in a neural network. Determining the optimal values for hyperparameters can be a time-consuming and computationally expensive process and requires extensive experimentation and testing.

At the core of the pipeline, the Automatic Training, Inference & Standardization module is responsible for building a model based on preprocessed data and the optimal choice of hyperparameters. Once a model has been trained, it can be deployed to a production environment where it can process new data and generate predictions or classifications. Different serializations are possible based on the type of model that was trained. We mainly differentiate between deep learning models and regular machine learning models.

Deep learning encompasses the machine learning models built on deep neural networks, whereas machine learning models can be any AI models that learn from data. Because of the structure of our model, each time a model is instantiated and, subsequently, trained, it can be directly converted to ONNX (for deep learning models)[10] or PMML (for machine learning models). Models are stored in the Knowledge Repository for future reuse.

Towards the end of the pipeline, we find the application of explainable artificial intelligence (XAI). Its application is of utmost importance as it addresses key challenges in AI adoption. XAI enhances transparency, fostering trust between users and AI systems by providing insights into the decision-making process. Once the model has been trained, the Explainability Generation module generates local and global explanations for all the models in the form of feature summary statistics, feature summary visualizations, and intrinsically interpretable models. It is worth noting that its application is important for promoting transparency, trust, and fairness in AI systems and for enabling researchers and practitioners to make better decisions based on insights from their data.

In order to ensure that the models can be deployed in any environment and are easily reproducible, the Edge Embedded AI Kit encapsulates AI pipelines in Docker images that are subsequently stored in a Docker registry and executed by the Model Orchestration component (see Section 4.2.4). In order to upload and download images, the Edge Embedded AI Kit has an API with push and pull methods, respectively. Those methods have been implemented using the Docker SDK for Python[11].

### 4.2.3. AutoML Algorithm Selection/Recommendation

The AI Model Generation pipeline (see Section 4.2.2) incorporates an automated process of recommending machine learning methods [53,54]. This recommender module aims to assist experts in selecting an optimal approach called strategy (an appropriately parameterized machine learning algorithm) for a given problem (task) by taking into consideration its inherent characteristics. These recommendations are prepared in a hybrid way, combining the advantages of two components:

- **Ontological Component**: Encapsulates AI methods and manufacturing domain knowledge using extended AI model ontology[12]. It establishes a shared conceptual framework enhancing the reliability and explainability of the recommendations.
- **Reasoning Component**: Employs inference rules to conclude strategy details (utilizing insights from both the Ontological Component and the Case-Based Model Repository) and initiates the model training phase in the AI Model Generation pipeline (see Section 4.2.2).

The recommendation process is initialized by establishing meta-knowledge based on information about previous machine learning experiments and their results. It includes data descriptors (meta-features extracted from stored datasets), task descriptors (task details), applied strategies (machine learning methods that, historically, are used to solve a considered task), and performance metrics (evaluation methods computed for employed strategies for example, precision). When a new task appears, meta-data for the new training set is calculated. Then, the meta-data is matched to the existing meta-knowledge. As a result of this matching, a recommendation is created containing the most appropriate learning algorithm for the task being considered, along with the selected parameters.

The entire recommendation process is collectively executed by four modules: Data Descriptor Filter, Strategy Generator, Strategy Ranker, and Model Filter. The Data Descriptor Filler module computes and updates the necessary dataset meta characteristics, which is crucial for the subsequent component: the Strategy Generator. While some descriptors, such as the dataset identifier and basic dataset features (e.g., a number of instances), might be provided by the user, additional dataset meta-characteristics require computation (e.g., standard deviation). According to the meta-features extracted and the task type, several types of data descriptors are found, including general, statistical, and conceptual, based on information theory and clustering meta-features.

The Strategy Generator is tasked with creating a comprehensive list of fully instantiated strategies, incorporating the most appropriate AI techniques and algorithms applicable to a specific dataset for addressing a designated task. The first step includes employing a graph-based inferential reasoning step to navigate the ontology, deriving potential strategies while considering the given task, data descriptors, and other pertinent aspects. Simultaneously, the case-based model repository is accessed through a CBR mechanism[13], acquiring historically effective models used for similar problems characterized by data descriptors and the given task. These analogous problems contribute additional strategies to complement those obtained from the ontology. Ultimately, all strategies are combined and can subsequently undergo filtration based on established information regarding the pending task.

The Strategy Ranker is responsible for generating the definitive list of strategies by taking into account certain aspects pertaining to the performance of AI models. The feedback on the performance of all strategies generated by the strategy generator comes from the AMG (see Section 4.2.2) component. The AMG component executes diverse strategies with varying hyperparameters to compute performance metrics that are stored in a local model repository. The Strategy Ranker module analyses performance data for each potential strategy, considering factors such as execution cost or other pre-set KPIs to finally rank the strategies based on the obtained performance metrics. These rankings are then presented to the user for evaluation.

The last module is the Model Filtering module, which governs the addition of new cases to the Case-Based Model repository. It determines the relevance of newly trained models, along with their characteristics and performance-related data, to be incorporated into the local repository of the Recommender System, adding the benefit of an automated learning mechanism.

### 4.2.4. Model Orchestration

In knowlEdge, we use CEML as a starting point for defining the requirements for creating an innovative and robust platform for the orchestration of the AI learning pipeline within the compute continuum. The primary objective of the Processing and Learning Orchestrator (PLO) component and its accompanying Deployment Agents (DAs) is to equip the high-level knowlEdge components with suitable tools and APIs for MLOps, thereby facilitating a seamless AI lifecycle orchestration experience for end users.

The architectural representation depicted in Figure 7, where boxes in grey represent Orchestration components, purple boxes represent other knowlEdge components, and finally, blue boxes represent interfaces, illustrates how the knowlEdge platform's components can deploy AI models and apply intelligence across diverse locations, spanning from network edges to cloud environments in the computing continuum. This process is accomplished through interactions with the PLO component, which efficiently delegates the AI model deployment requests coming from the other knowlEdge components to the most fitting DA. When a component requests the deployment of a model through the Core Orchestrator subcomponent's REST APIs, the PLO module automatically selects the most appropriate DA based on factors such as node computational power, latency requirements, and data locality. Importantly, the PLO and DA components are designed to operate within different environments and platforms, including edge devices, on-premises servers, and cloud-based infrastructures. This enhanced automation in the deployment process minimizes the need for manual intervention and optimizes the deployment of AI models throughout the edge-to-cloud continuum. The secure transmission of deployment processes across the network is facilitated by a dedicated Transport Layer Security (TLS) relay server and clients, establishing an isolated tunnel. The Agent module initiates the deployment process within the desired environment, transmitting the request via an asynchronous queue to a Worker module responsible for fetching the model images, which contain the actual models and the frameworks required for performing the training and inference operations, from the repository managed by the Edge Embedded AI Kit component (see Section 4.2.2).

In order to ensure isolation among potential instances of DAs residing within the same environment, the containers holding the model images are deployed inside a dedicated Docker-in-Docker (DIND) container[14]. Each model container, once deployed, can exchange data with any other knowlEdge component through the message broker (see Section 4.1.2) by using the MQTT Interface.
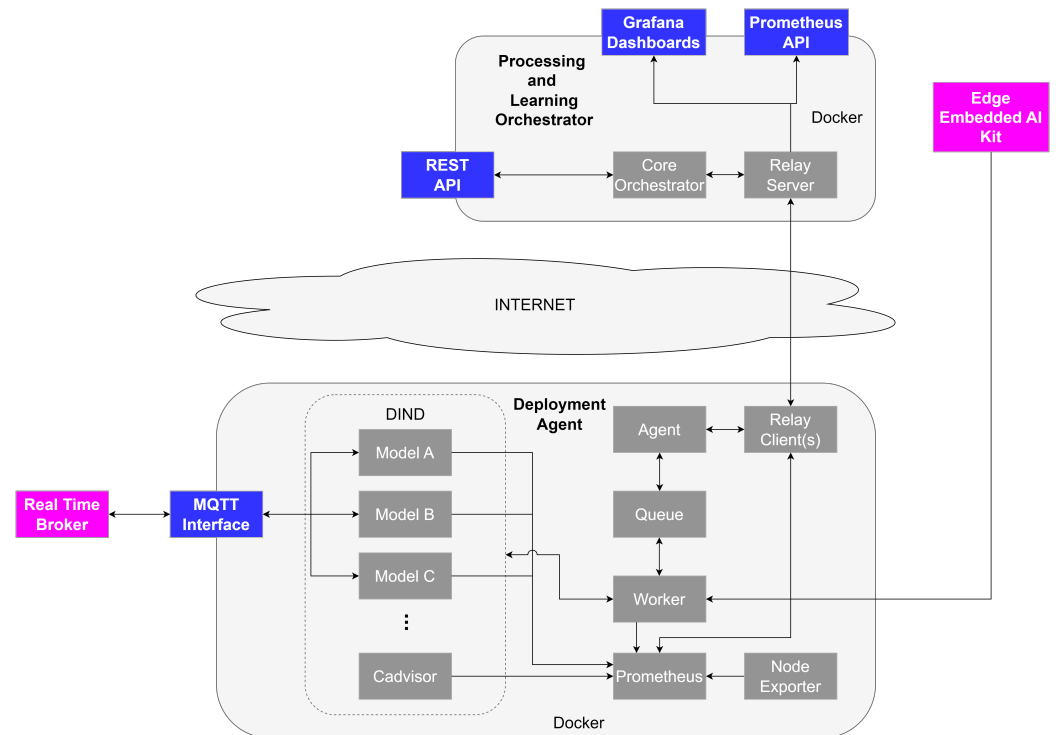


**Figure 7.** Orchestration components architectural schema. Grey boxes represent Orchestration components, purple boxes represent other knowlEdge components, and blue boxes represent interfaces.

The DA module monitors the resource utilization of each deployed model by leveraging a dedicated Prometheus[15] instance. This provides real-time performance information to operators through a user-friendly Grafana-based dashboard[16], enabling them to promptly assess the efficiency of infrastructure utilization. These metrics can also be accessed by any other knowlEdge component by querying the Prometheus APIs exposed by the PLO. The integration of Prometheus, cAdvisor[17], and the Node Exporter modules[18] within the component equips operators with comprehensive real-time hardware monitoring and logging capabilities, empowering them to improve the effectiveness of the orchestration process by retaking manual control of the process at any time, keeping the human in the center.

## 4.3. Human-AI Collaboration

In knowlEdge, there is a special emphasis on how humans interact with the AI models generated by the knowlEdge Discovery component (see Section 4.2.1) and the AI Model Generation component (see Section 4.2.2). Developing a framework of AI solutions that capture and process data from various sources, including human-AI collaboration [55], is one of the key challenges in modern-day agile manufacturing [56]. This challenge is exacerbated by the lack of contextual information and nontransparent AI models.

The purpose of human feedback is to provide a user interface to allow subject matter experts to inject domain knowledge into AI models in order to provide semantic information to previous knowledge, as, for example, a detailed description of a process or data. It offers enhanced comprehension of the data, and it also enables a more thorough evaluation of the entire AI pipeline. In other words, the role of human-AI collaboration is to offer an interface between subject matter experts and AI models, where, in this case, the goal of the

human-AI collaboration is to facilitate human feedback for domain experts, i.e., machine operators and managers who may not have deep knowledge of the intricacies of AI models.

After reviewing prior research, we describe our concept domain knowledge fusion in agile manufacturing use case scenarios for human-AI interaction. We identify two kinds of knowledge: (i) learned knowledge, i.e., the knowledge generated by the AI model, and (ii) engineered knowledge, i.e., the knowledge provided by the domain expert. We identify three aspects of domain expert interaction with our AI systems to observe and (i) reject if the learned knowledge is incorrect, (ii) accept if the learned knowledge is correct, or (iii) adapt if the learned knowledge is correct but needs modification. We demonstrate these concepts for researchers and practitioners to apply human-AI interaction in human-centered agile manufacturing.

As shown in Figure 8, where boxes in white and grey represent Human-AI components, Human-AI Collaboration is composed of multiple subcomponents and interfaces that enable communication with external systems, such as data sources, model repositories, machine configurations, and decision support systems. The subcomponents are (i) Model and Data Selection, which enables operators to choose AI models and data from a list of available options in order to evaluate the effectiveness of the AI models on the given data for the specific scenario at hand; (ii) the subcomponent for Parameter Optimization provides managers and machine operators with an interface via which they can select the parameters in order to try various possibilities of values and observe the outputs in order to implement parameter optimization that could lead to the best outcome for the scenario at hand; (iii) the Configuration Adaptation subcomponent enables machine configurations or measurements to be updated or upgraded, and in cases where a model necessitates specific machine configurations that require modification, operators/managers can adjust the machine settings to align with the relevant model. For example, if new machines or sensors may need to be included within the Human-AI Collaboration ecosystem, their configuration should be incorporated and stored in such a way that they are accessible to the modules which, in turn, ensures extensibility; (iv) Domain Ontology Enrichment with engineered knowledge describes the scenario where the AI model analyzes the given data for a task (e.g., outlier detection) and produces its result (e.g., that a given data point is an outlier), the domain expert realizes that the output of the model is not right (e.g., that the data point is not an outlier), and the information provided by the domain expert (i.e., the data point is not an outlier) is stored in the repository of ground truth and sent back to the AI model for retraining. It can be used by operators and managers to enrich the knowledge repository by incorporating new entries derived from executing the system with diverse settings for models, parameters, and configurations.

Figure 9 demonstrates the implementation of the Human-AI Collaboration process, which involves the selection of data and models, as well as parameter optimization. This includes the data flow and user interface (UI) for model selection and parameter optimization. Through this UI, domain experts can choose models, set parameters, and optimize parameter values. As indicated with a red star in the model and data payload, the model requires availability of data url and a pre-processor corresponding to the selected model.

The UI displays a visualization of the processing results based on the chosen model, parameters, and values. After the expert finalizes a decision on which model, parameter, and values to proceed with, the UI then provides an option to export the results, which are subsequently utilized by the Decision Support System (DSS).

Moreover, the domain expert selects a section of the visualization and provides engineered knowledge, i.e., the manual labeling of data points. This helps the user to visually inspect the dataset and enrich it with domain knowledge to boost the quality of the data to be used as a training dataset for better ML model performance. For example, for an AI model built for anomaly detection, this is achieved by enabling the user to select the data point on the visualization plot in order to display and review (and, when applicable, modify) the data that are marked by the system as anomalies.
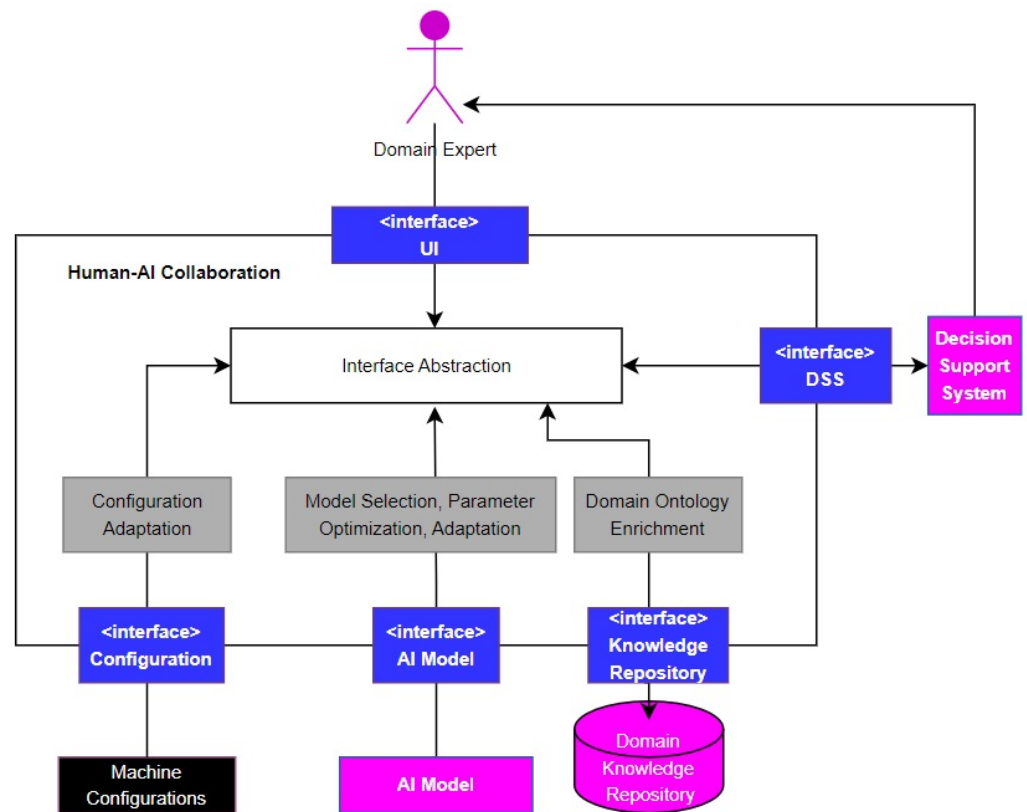
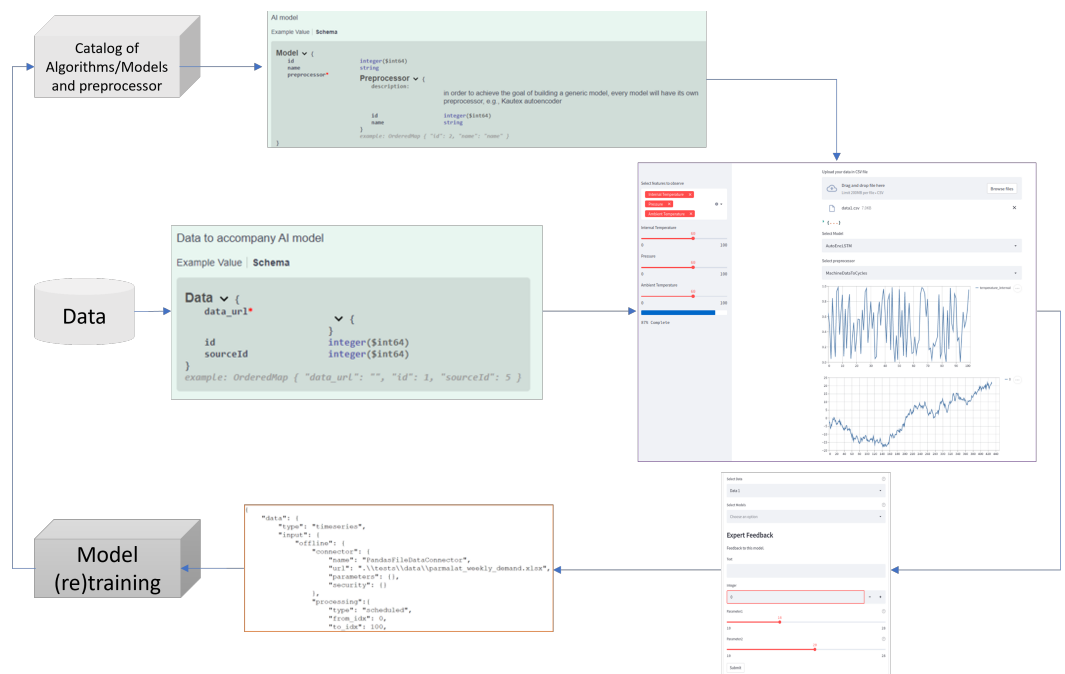**Figure 8.** Human-AI Collaboration components and Interfaces.



**Figure 9.** Human-AI Collaboration Flow Diagram.

*4.4. Model Sharing-knowlEdge Repository*

The knowlEdge Repository is the central cloud instance used to store AI models and their describing meta-data in the project. This component is used for model deployment since it allows a feature rich access to former results and models for the development of future models, as well as for the use of already existing models generated by other

components (see Section 4.2). This information is designed in order to find similar models and to give the decision support system as many opportunities as possible to suggest appropriate models. Despite this, there is still a high degree of flexibility to account for unusual models or innovative evaluation metrics.

### 4.4.1. Technical Overview

The knowlEdge Repository is designed to be available as a single cloud instance that is accessible by a REST-API. It can be containerized using Docker [51] and consists of three parts (see Figure 10, where grey boxes represent knowlEdge repository components, purple boxes represent other knowlEdge components, and blue boxes represent interfaces):

- **knowlEdge Repository Management**: The server hosting the API (implemented in Python [57] using Flask [50]);
- **Metadata Database**: A NoSQL database using MongoDB [58] to store the meta-data in an efficient manner and ensure fast access;
- **Model Database**: A Hadoop-distributed file system [59] that stores the model files in a distributed and accessible way. The models must be in the ONNX[19] or PMML [60] format.
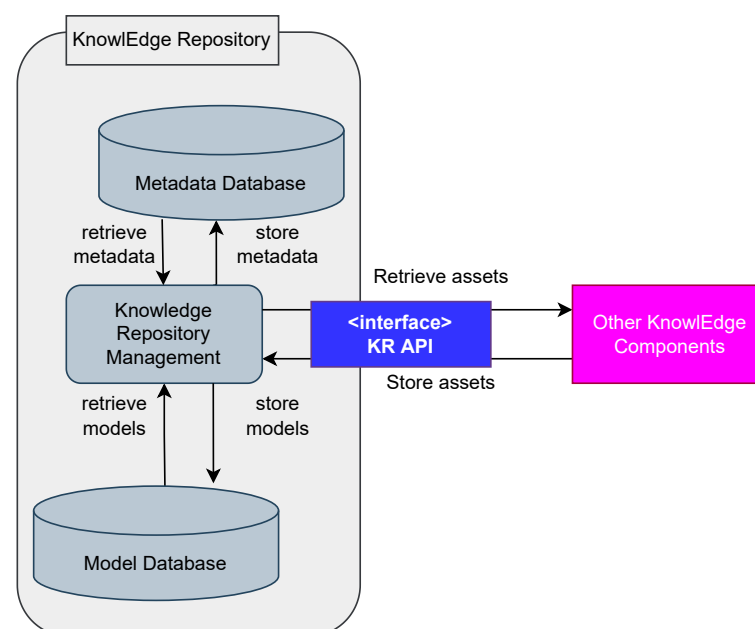


**Figure 10.** The architecture of the knowlEdge Repository.

The meta-data is stored using a fixed ontology (see Section 4.4.2 and Figure 11) to ensure a feature with rich reusability of the models, while the model files themselves may consist of ONNX or PMML files. This way ensures that the models stored in the knowlEdge Repository can be used in many different machine learning frameworks. While PMML is an apt format for many classical machine learning techniques and pipelines, it is not suited for models with a large parameter count, such as with neural networks. In order to circumvent this problem, a second format, ONNX, is accepted, as this allows for storing even big neural network models in a memory-efficient fashion.

### 4.4.2. knowlEdge Ontology

In order to ensure the meaningful identification of fitting models for different tasks, a plethora of meta-data is stored in the knowlEdge Ontology (see Figure 11). It is based on the ML schema ontology [61] but does not include some information that is not useful or is even misleading in our context. In total, 12 different types of entities exist in the knowlEdge Ontology, which are described in more detail in the following paragraphs. These can be split into *user-related*, *model-related*, *task-related*, and *performance-related* entities.

The only **user-related** entity type is the *User*, which is needed to store the models in a meaningful way. It consists of a name, an affiliation, the time of its creation, and an e-mail address. A User can be the owner of multiple Models and can also create Application, Data, and Property Type entities.
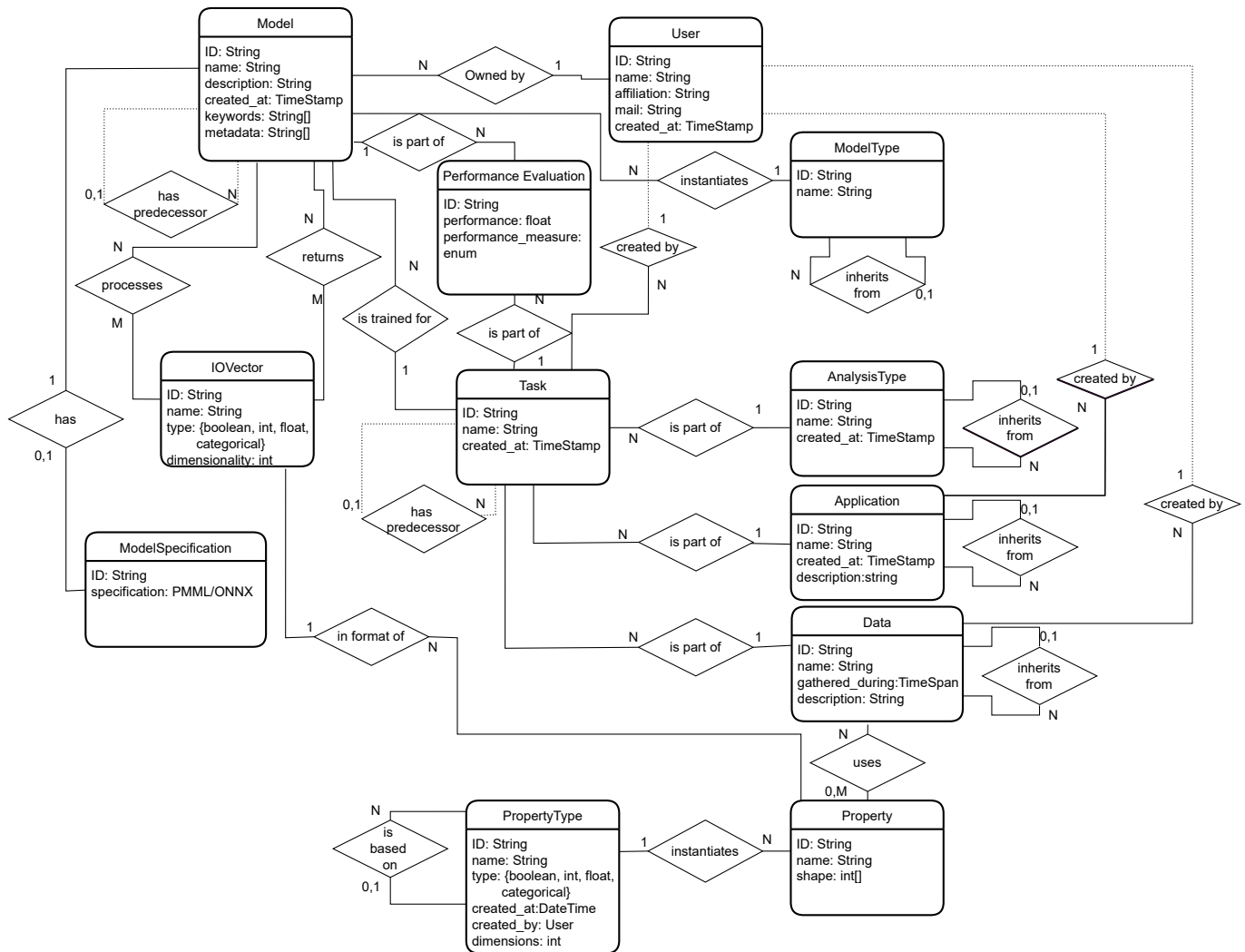


**Figure 11.** The Ontology of the knowlEdge Repository.

The **model-related** entity types are Model, Model Specification, IO Vector, and Model Type. These contain information used for the Model Deployment. The *Model* resembles the most important information about a Model, such as its name, description, and the time it was created. A Model is also connected to a Model Type and a Model Specification. Models can hierarchically have children or parents and are also linked to several IO Vectors as in- and output. A model is always trained for a single Task, while it can be linked to other Tasks with several Performance Evaluations. A *Model Specification* is the actual model used for Deployment. It can either be specified as a PMML or an ONNX file, specifying the model. *IO Vectors* are used to represent the in- and output of a Model. They contain a name, dimensionality, and data type. These can be in- and output to several Models. They can also specify a pipeline of Models, where the output of one model is used as the input of a subsequent Model. A *Model Type* is used to categorize different Models so that a User can detect similar Models. In the data schema, this is only presented by its name and the hierarchy between different Model Types.

The **task-related** Entity Types resemble the information from Problem Definition, Data Acquisition, and Data Preparation. A *Task* has a name, the time it was created, an Analysis

Type, an Application, Data, and a possible parent model. Additionally, the User who created the Task is stored. Several Tasks can be children of a single Task, and multiple Models can be trained for any Task. A Task, hence, is the complete description of a problem, which concludes all information needed by a Model. An *Analysis Type* resembles a formal viewpoint for a problem (e.g., image recognition). These can also have a hierarchy to give granular possibilities to classify a problem. It is only represented by its name. Similarly, an *Application* is the business-oriented view of a problem (e.g., defect detection). A *Data* entity consists of the concrete dataset gathered during Data Acquisition and Data Preparation. It contains its differing Properties as well as the timespan over which the data was acquired. A *Property* resembles a single measurement and consists of its shape and the Property Type it implements. These *Property Types* consist of the Type of the data as well as the number of dimensions each property has.

The **performance-related** Entity is the *Performance Evaluation*, which is used for Model Evaluation. It represents a measurement of a given performance measure from a Model on a Task.

This information can help to identify similar problems and identify already existing solutions for given problems. In particular, the hierarchical entities enable feature-rich meta-data to identify fitting models for a given problem.

*4.5. Data Generation and Model Validation-Digital Twin Framework*

Digital twins (DTs) are virtual representations of objects, products, equipment, people, processes, or even complete manufacturing systems. Digital twins are used to improve operational efficiency, predict maintenance needs, and optimize performance [62]. Digital twins play important roles at different stages of an AI system's development.

In Data collection and labeling, digital twins generate synthetic data that augment real data to train AI models.

When training the Model, digital twins provide realistic environments to develop, test, and optimize AI models. DTs can provide feedback and refinement, monitor the performance that AI models have in the real world, and provide data to continually improve the models. Finally, while evaluating and optimizing models, digital twins can simulate complex scenarios to find the best ways to deploy AI systems and integrate them with human workflows.

In recent years, digital twins have evolved from isolated representations (mainly CAM models in the design phase) to core integrations of other solutions, such as AI models, IoT data frameworks, robotic controllers, or advanced AR systems. Nowadays, digital twins play an integral role in improving manufacturing efficiency and process optimization, as they represent a counterpart of the physical reality and allow for the integration of existing relevant technologies, such as machine learning solutions, visualization analytics dashboards, and data transformation pipelines. A digital twin can capture the entire lifecycle of a product or process, from design and development to operation and maintenance [63]. This allows for continuous improvement and feedback loops across different stages and stakeholders. However, the normal approach is to have specific digital twins at different lifecycle phases or for specific domains, as the set of functional services required is different. Digital twins can communicate with their physical counterparts in real time, providing alerts, insights, and recommendations based on data analysis and artificial intelligence while controlling them as a result of computing machine learning models with smart decision-making actions. Digital twins are at the core of the next evolution of AI algorithms, as manufacturing digital twins provides the mechanisms to generate realistic synthetic data that permit machine learning solutions to learn from scenarios where data are not available or where getting or evaluating data is very expensive [64].

The knowlEdge Digital Twin functional architecture is presented in Figure 12, where boxes in white represent UI interfaces, grey boxes represent backend digital twin components, purple boxes represent other knowlEdge components, blue boxes represent inter-

faces, and finally, black boxes represent external components. The main functional building blocks are covered here:
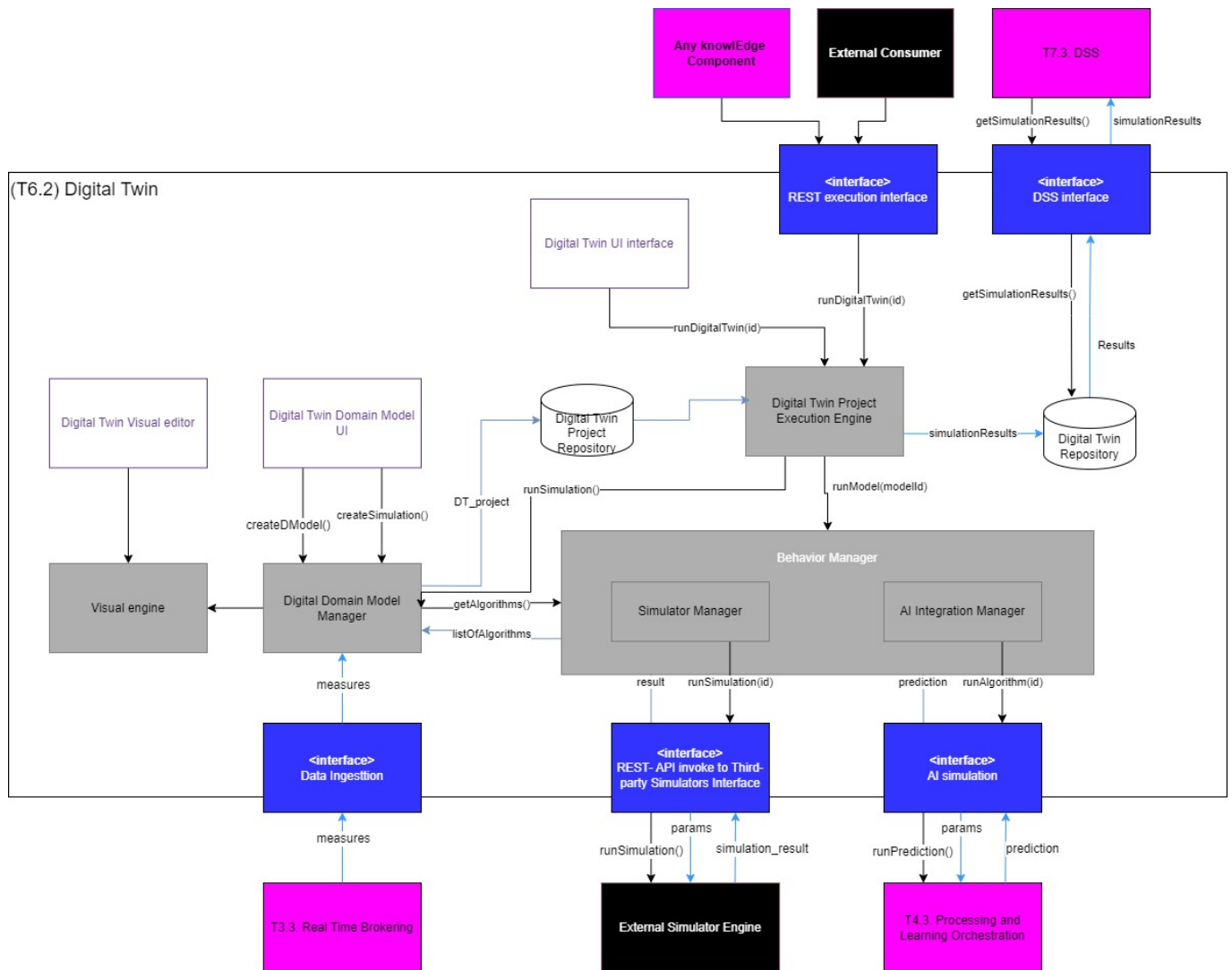


**Figure 12.** Digital Twin architecture framework.

- State Management: Digital twins register the information of physical assets, products, and processes, among others. The knowlEdge Digital Twin covers the editing of digital twin data models with prepopulated concepts rooted in ISO 23247 and following the JSON-LD format, making the solution compatible with other solutions, such as FIWARE context brokers. Models are populated thanks to continuous synchronization with the real-world data provided by the Data Collection Platform (see Section 4.1.1).
- Digital Twin Visual editor: Many digital twins cover topics in the design phase, where uncertain or never-before-seen scenarios are simulated and put under stress; knowlEdge provides a 3D editor and viewer based on web-based solutions such as three.js[20]. The visualizations are also used to support control in the production phases of shop floors.
- Behavior Simulator Manager: Supporting the integration of internal and external simulators. The knowlEdge platform provides a range of simulators in areas such as production scheduling, resilience evaluation, simulation of physics based on stochastic processes, and the generation of synthetic data based on AI copulas algorithms. Digital twins interact with the AI models generated by the AI model generator (see Section 4.2.2) to run AI simulations and surrogate AI models when traditional physics-based simulations are not computationally efficient. Digital twins are an inte-

gral part of the lifecycle of AI/ML solutions, as the real or simulated data generated by their simulations can be the perfect input for training ML models or running inferences from those ML models. On the other hand, the predictions generated by AI models can be validated through shop floor simulations by using the digital twin.

## 5. Use Cases for the knowlEdge Framework

The development of the knowlEdge project was motivated by specific use cases and practical applications within the manufacturing sector. Three prominent use cases, each representing a distinct manufacturing context, have been identified: Parmalat, Kautex, and Bonfiglioli.

### 5.1. Parmalat: AI-Optimized Production Planning in the Dairy Industry

Parmalat, a major player in the Italian milk market and a central focus of the knowlEdge project, is undergoing transformative advancements at its Collecchio and Rome plants. Collecchio, the flagship facility, focuses on the intricate task of scheduling packaging lines efficiently, where the development of a scheduling tool capable of dealing with finite capacities can be very beneficial. This tool addresses the challenge of aligning production with market demands and resource constraints, with the overarching goal of optimizing sequential processes, enhancing efficiency, and minimizing losses through the integration of AI technologies. Simultaneously, the Rome plant aims to extract and integrate information from diverse data streams to predict requested volumes, optimize warehouse management, reduce stock, improve production flows, minimize waste, and enhance overall co-ordination.

The impact of these use cases on the project's design is profound, emphasizing the imperative for AI-driven solutions adept at handling complex production scheduling and dynamic adjustments. The project strives to provide tools capable of predictive simulations, minimizing human input, and enhancing the prediction of quality parameters in incoming milk. The scenarios underscore the critical importance of real-time rescheduling, efficient data utilization, and the ability to adapt swiftly to changes in demand or unforeseen disruptions, shaping the project's vision for a comprehensive and responsive AI-powered manufacturing framework.

### 5.2. Kautex: Understanding Process Parameters in Automotive Manufacturing

Kautex, a key player in the automotive industry, focuses on understanding the convoluted dependencies between process parameters affecting product dimensions in plastic fuel system manufacturing. The project aims to develop a system that supports technicians and specialists in setting up procedures, predicting the impacts of different settings, and receiving immediate adjustments or countermeasures in response to detecting deviations during the sampling and production processes.

The Kautex use case underscores the importance of understanding the interactions among an abundance of parameters influencing product attributes. It influences the design by emphasizing the need for a system that facilitates the setup of procedures, highlights potential impacts, and motivates quick reactions to deviations. The project aims to develop AI models capable of enhancing the learning process from past issues and settings, ensuring a more robust and adaptive manufacturing process.

### 5.3. Bonfiglioli: Automation of Assembly Quality Control

Bonfiglioli, which specializes in power transmission and drives, targets the improvement of assembly quality and product reliability. The focus is on automating quality controls during the assembly process, reducing failures, and ensuring a safer and more usable workplace for operators.

The Bonfiglioli use case highlights the importance of quality control automation in assembly procedures. The project aims to develop systems that not only improve assembly quality but also enhance operator safety and usability. The scenarios emphasize the need

for real-time quality checks, immediate troubleshooting, and comprehensive data recording for continuous improvement.

The knowlEdge project's design has been significantly influenced by these specific use cases in the manufacturing sector towards AI-driven solutions capable of handling complex scheduling, dynamic adjustments, and real-time adaptability. The emphasis on predictive simulations, the reduced need for human input, and improved quality predictions directly stem from the requirements of these real-world manufacturing challenges.

## 6. Platform Design and Deployment

The platform's design process involved a systematic approach, beginning with the identification of specific use cases within the manufacturing sector. By engaging with partners such as Parmalat, Kautex, and Bonfiglioli, the project gained valuable insights into diverse manufacturing contexts, allowing for the identification of unique challenges and requirements. The design phase meticulously crafted integration workflows, ensuring seamless collaboration between AI models, human experts, and manufacturing environments. The development team prioritized adaptability, collaboration, and efficiency to address the complexities of real-time adjustments, data diversity management, and human-AI collaboration.

The resulting architecture diagrams provide a clear framework for the deployment phase, with specific attention to cloud-based training, inference in fog, and other scenarios tailored to the diverse needs of the manufacturing partners. In order to ensure seamless deployment, two distinct environments were defined: a testing environment in LINKS[21] and a production environment for each of the pilot cases. The testing environment played a pivotal role in the deployment strategy, serving as a controlled space to validate the platform's functionalities, troubleshoot any potential issues, and fine-tune configurations before actual implementation in the pilot premises.

Once on the pilot premises, two distinct deployment scenarios based on their specific needs were defined.

### 6.1. Training Cloud-Based/Inference Fog-Based

In this scenario, AI models undergo training in the client's cloud infrastructure, leveraging the computational capabilities of a remote environment for tasks such as processing extensive datasets and training complex models. Following the training phase, the models are deployed and executed on fog devices or edge nodes, facilitating real-time inference within the manufacturing environment. The fog nodes, located closer to edge devices, handle local data for efficient inference tasks. This deployment strategy offers advantages such as efficient resource utilization, handling large-scale datasets, and flexibility for accessing tools and libraries for model development. Centralized management in the cloud streamlines model training, deployment, and updates.

### 6.2. Training and Inference in Fog

In this alternative scenario, both training and inference occur directly at fog or edge devices. Fog nodes possess sufficient computational resources to handle training tasks, and the trained models are executed locally on the same devices.

This scenario prioritizes low latency and offline capability, as data processing and decision-making happen locally without relying on cloud communication. It enhances privacy and security by keeping sensitive data on local devices and reducing the need for external server transmissions. However, challenges arise from the limited computational resources of fog devices, especially for complex and large-scale training tasks. Data remain on the edge device, minimizing transmission needs and reducing latency, with only model updates transferred between the edge device and the server, significantly reducing bandwidth requirements. This approach empowers fog devices to contribute to the training process effectively.

The first strategic deployment approach entails distributing various components across the continuum: the edge, fog, and cloud. Consequently, the components responsible for data acquisition and preparation are strategically placed across all three layers to ensure universal data accessibility. Regarding the learning process, critical tasks, such as feature extraction or anomaly detection, are performed in the fog layer. This decision is rooted in the idea that executing these tasks closer to the data source minimizes latency and augments real-time processing capabilities. The recommendation and model generation components are flexibly deployed in both the cloud and fog layers, depending on the specific action—whether it involves training or inference, akin to the orchestration process. Additionally, components associated with human-AI collaboration and the user interface find exclusive deployment in the fog, capitalizing on benefits such as reduced latency and an enhanced user experience. Lastly, the knowledge repository's deployment in both the fog and cloud facilitates updates during both the training and inference phases. It is crucial to note that the knowlEdge Marketplace is deployed solely in the cloud, serving as a centralized hub for knowledge management, fostering collaboration, and efficiently disseminating knowledge within the project's domain. This cloud-based knowlEdge Marketplace acts as a central point for collaborative efforts and streamlined knowledge sharing throughout the entire project ecosystem.

In cases where only the fog layer is available, all the components are deployed in the same layer.

Considering both deployment strategies, Parmalat and Kautex are set to follow the deployment scenario outlined in the first case, involving cloud-based training and inference in the fog. This approach ensures the efficient utilization of computational resources and centralized management in the cloud, allowing for large-scale model training and streamlined deployment processes. However, the decision to adopt this scenario is not solely based on technical considerations; privacy concerns also play a crucial role. By keeping training processes centralized in the cloud, Parmalat and Kautex can manage sensitive data more securely and efficiently. In contrast, Bonfiglioli, with a focus on privacy and security in assembly quality control, adopts the second case, involving both training and inference in the fog. This approach prioritizes local processing and decision-making, minimizing the need for transmitting sensitive data to external servers. In essence, the deployment choices align with both technical requirements and the specific privacy considerations of each manufacturing partner.

## 7. Discussion

As it is perceived, AI lifecycle orchestration can be strictly connected with Industry 5.0 principles as long as it is based on a compute continuum with the aforementioned characteristics. In the current work, we have defined the fact that an edge-to-cloud continuum is able to meet both the requirements (a) coming from companies, including those related to automated and streamlined AI/ML processes, and those (b) coming from the European Commission related to ethics and trustworthy AI[22] with Human-in-the-Loop as a core element of Industry 5.0. In particular, by applying the knowlEdge framework for human-centric AI lifecycle orchestration, we contribute to the following human-AI lifecycle stages:

- Starting with the data acquisition part, as soon as data are collected, data exploration starts. At this stage, based on the introduced knowlEdge Discovery Engine (see Section 4.2.1), a human can be involved in the feature extraction process.
- Regarding model development, humans can be involved by using the Human-AI Collaboration module (see Section 4.3) that has been developed. By using this component, an end-user can provide information regarding the data source, the task, its setup, and the final strategy to be followed regarding execution and deployment.
- In order to further assist humans in the selection of AI models that should be executed in the task at hand, an AI system should be able to provide a recommendation strategy and XAI functionalities to the user at the same time. These requirements have been

covered in knowlEdge, following the approach described in Section 4.2.3. A recommender based on semantic reasoning provides recommendations about which strategy should be followed and presents the AI model's performance metrics. Information regarding the importance of features/variables during the models' training is also available by using libraries such as Dalex.

- In order to further boost human-AI collaboration regarding model execution, a user is able (by using UIs) to select models and parameters and to optimize the model's parameters based on his/her knowledge as a domain expert.
- The continuous monitoring of AI model performance and maintenance can be incorporated by AI pipelines. In knowlEdge, such functionality has been added and extended again with human-AI features, keeping humans in control of retraining the AI models. In order to support this functionality, a kind of knowledge base for training models is needed in order to keep previous versions of the models alongside various meta-data, including the AI model's metrics. For this purpose, knowlEdge has deployed a repository that is based on an ontology for describing AI model meta-data.
- AI Bootstrapping and testing of AI model performance can be improved by using simulation environments. For the knowlEdge case, digital twins can simulate manufacturing environments and validate the results of the AI models in a virtual environment.
- Finally, UIs are of great importance to support human-centric AI orchestration. knowlEdge features user-friendly interfaces capable of explaining AI outcomes and input controls, along with navigational elements, to enable users/experts to insert their feedback and expertise during various stages of AI lifecycle orchestration.

The proposed knowlEdge approach moves a step further from current AI orchestration approaches, as it introduces strong human-AI collaboration during various stages. Several cloud orchestration solutions have become available, such as Kubernetes[23], Cloudify[24] and Docker Swarm[25]. Major cloud providers also offer tools for orchestration, such as AWS CloudFormation[26], and Google's Deployment Manager[27]. However, they do not cover the cloud-to-edge orchestration that knowlEdge introduces, and this better meets the needs of smart manufacturing domains, as data are available both on the edge and in the cloud. In recent years, the orchestrators considering edge resources have been enabled by open source frameworks such as KubeEdge [65]. However, the current solutions struggle to manage the dynamism (at application and infrastructure) in edge and fog computing environments, as they are primarily inspired by the reliability inherent in cloud environments. Other approaches, such as MiCADO [66], introduce an orchestrator for the cloud-to-edge continuum; however, this focuses on the delivery of an operational solution. The knowEdge differential approach adds the human-centric concept as a part of AI orchestration in the cloud-to-edge continuum.

In addition to the orchestration part, the introduced solution contributes to the Industry 5.0 domain by providing a complete framework for data collection, analysis, forecasting, and decision support, with a strong focus on the human aspect. Early approaches related to the selection of various AI models to solve an industrial project have been introduced by several EC projects. In Boost 4.0, a cognitive analytic platform powered by AI solutions [67] was introduced. It provides model selection based on various metrics. However, the metrics are used for auto-triggering retraining without human collaboration and XAI functionalities are missing as well. In general, there are limited solutions regarding the delivery of AI services that are fully relevant to Industry 5.0 concepts. In general, most of the articles related to Industry 5.0 mainly focus on the concept itself, its trends [68], available technologies towards Industry 5.0, [69,70] etc.

In addition to the research outcomes, ICT world leaders have delivered various cloud-edge IoT platforms that provide AI solutions for smart manufacturing. The SIEMENS Edge Computing Platform[28] for the Machine Tool domain combines SIEMENS Mindsphere[29] cloud services with an edge runtime environment. This approach enables the development and deployment of applications for the edge environment. However, it is limited to the machine tooling domain, and the AI applications that support inference and analysis

are not clear. A similar approach was introduced by SAP—SAP Edge Services[30]. It is considered to be an extension of the core SAP Analytics Cloud[31]. However, the edge part seems more focused on data streaming and preprocessing, whereas services such as training, analysis, and inference are on the cloud. Microsoft's Azure IoT-edge[32] provides a solution for customers to deploy models that are built and trained in the cloud and run them on-premises. AWS for the edge[33] from Amazon is another alternative that enables data processing, analysis, and storage close to customer's endpoints, allowing for the deployment of APIs outside of AWS data centers. As in the case of the knowlEdge approach, the ones provided by world leaders are based on the containerization approach. Their pros are that they are widely used and tested, they support various communication protocols and data types, and they have extensive documentation and support. On the other hand, they are very generic, as they aim to meet customer needs beyond smart manufacturing, and they lack human-AI collaboration approaches to Industry 5.0. knowlEdge (to the best of our knowledge) proposes one of the first complete platforms for smart manufacturing that is able to support complete AI lifecycle orchestration for the delivery of human-centric services in the Industry 5.0 era.

Furthermore, pathways regarding artificial intelligence in manufacturing have been introduced. The adoption of AI practices by companies has been classified at various levels [71], starting from Level 1, where no AI is involved and humans are in control, and reaching Level 5, where AI is in control and no humans are involved in decision-making. In the case of knowlEdge, which focuses on industry partners with limited experience in AI, all use cases begin in their original state at Level 1. This means that humans are fully in control, and no AI systems are involved in the decision-making process. However, the implementation of the technologies covered in the article leads the demonstrators to advance along the AI pathway. Although there is not a single level involved, the functionalities introduced are distributed across Levels 2, 3, and 4. This includes AI assistance, where AI systems provide additional information to aid human decision-making; AI recommendation, where the AI suggests decision options for human evaluation; and collaborative AI, which involves a collaborative approach to decision-making with mixed teams of humans and AI systems. It is important to note that knowlEdge intentionally avoids reaching Level 5. This level implies that AI is in complete control without human involvement. The work aims to maintain human participation in the manufacturing setting, prioritizing collaboration between humans and AI rather than the full autonomy of AI systems in order to remain fully compliant with EC guidelines and ethics.

Finally, we summarize the results of our described component designs as follows:

- Human-AI Collaboration: The knowlEdge platform places humans at the core of the whole AI pipeline, from development to deployment. This collaboration enhances the quality of AI models based on human expertise by allowing domain experts to collaborate effectively with the platform.
- Data Exploration and Management: The platform includes a comprehensive toolset for data acquisition and exploration. The knowlEdge Discovery Engine empowers users by allowing them to extract meaningful features from raw data.
- Model Selection and Development: knowlEdge introduces a novel model recommendation system based on semantic reasoning, allowing users to participate in and understand the decision-making process to select suitable models for domain-specific tasks.
- Human-AI Feedback Loop: The platform natively integrates XAI techniques to provide transparent insights into the behavior of the AI models in production.
- AI Monitoring and Maintenance: The platform features a pro-active AI model monitoring system that detects performance degradation and allows for the automatic scheduling of model retraining, also taking human feedback into account.

Moreover, we showed three further key aspects through which the knowlEdge approach advances AI lifecycle orchestration:

- The knowlEdge platform presents a novel approach to AI lifecycle orchestration that is differentiated from other proposals by emphasizing human-AI collaboration and

placing it at the center of the process. This allows for human expertise in manufacturing domains to be brought closer to AI capabilities, which are usually too specific for AI experts.

- The integration of a Digital Twin Framework is a novel contribution, allowing domain experts to execute realistic simulations that facilitate the training, testing, and optimization of AI models way before bringing them into production.
- Another novel contribution is the addition of an AI model recommendation system based on semantic reasoning and explainability, enabling transparency and more user-friendly interfaces closer to the nuances of each specific manufacturing domain. This should enable manufacturing experts to understand and control the AI models better.

## 8. Conclusions and Future Work

In this work, we introduced the knowlEdge Platform as a software solution capable of assisting developers of AI-based solutions along the whole AI lifecycle. Through an emphasis on human-AI collaboration, it bridges the gap between domain experts and AI capabilities. The approach presented shows how the different knowlEdge development solutions perform data cleaning, data engineering, AI training, and development in the cloud-to-edge continuum, empowering the user with tools to enrich the AI models (AI-Human collaboration and knowledge management) semantically to support their training (and retraining based on domain expert knowledge) and to digitize the manufacturing assets through the knowlEdge digital twin. However, given the potential challenges related to varying technical expertise among domain experts, future research could investigate how the framework accounts for this diversity and whether additional training or user-friendly interfaces are needed. The integrated solution is currently being validated in three industrial companies in the dairy, automotive, and tooling manufacturing sectors. In the future, it will be essential to discuss how the framework can adapt to industries with different production processes, data structures, and regulatory environments. Exploring customization options or industry-specific modules could enhance its versatility. The AI solutions tested ranged from the cloud training of AI model solutions to deployment in the fog/edge through the Edge Embedded AI kit and the Processing and Learning Orchestration component. The solution improves the time span of AI solution development and supports the pro-active detection of performance degradation of AI models through the active monitoring of AI model results and the continuous collaboration of AI and humans. In this paper, we focus on presenting the human-centered AI orchestration framework rather than the results of specific pilot applications. Therefore, our documented findings are related to the core parts that a human-centered AI system for Industry 5.0 should include. The concept of Industry 5.0 aligns with a human-centric approach, recognizing the vital role of humans in the manufacturing industry. In addition, the integration of explainable AI techniques for transparency also takes into account the ethical considerations related to the interaction between humans and AI. This avoids potential biases in AI models. Thus, technology should indeed be seen as a supportive tool, augmenting human activities to improve processes and generate greater value. Consequently, the integration of human-AI interaction, exemplified by knowlEdge, focuses on key aspects of human-centered manufacturing, including understanding human behavior and reasoning, combining human and AI capabilities, and managing skills and knowledge. By embracing these approaches, we can create a future where humans and technology collaborate harmoniously, leading to more effective processes and increased value in the manufacturing industry. In addition to the digital twin technology, the framework can also evolve by incorporating emerging technologies, such as augmented reality (AR) or virtual reality (VR), to enhance human-AI collaboration. For instance, integrating AR interfaces for real-time data visualization during manufacturing processes might provide an immersive and interactive experience for users. Moreover, future work can focus on user-centric design and usability by incorporating user feedback mechanisms within the platform, involving regular usability testing, and providing customizable dashboards for a more user-friendly experience.

When examining the future implications of human-technology interaction, one notable aspect is the role of human-AI interaction in triggering participatory management. This means that the involvement of both humans and AI systems in decision-making processes allows for a more inclusive and collaborative approach to management, departing from traditional hierarchical management structures and embracing bottom-up or circular approaches. Furthermore, human-AI interaction, when implemented at the appropriate points or nodes within processes, facilitates self-steering. This means that employees have more control and autonomy in managing their own tasks and responsibilities within the overall manufacturing process. This can contribute to the development of more adaptable and resilient structures, ensuring a greater probability of continuity even within unforeseen events.

As it is perceived, it is essential to recognize that human-AI interactions can have far-reaching impacts that extend beyond the immediate routines of manufacturing processes. However, beyond the human-centered part that is covered to a high degree by the knowlEdge framework, the future steps of this work also include understanding the compliance of the framework with the other core trends (or event standards) of smart manufacturing, such as the concept of common data spaces and the knowlEdge sharing among organizations. The human-AI-centered AI orchestration framework that is introduced here will be extended to be compatible with data space concepts regarding sovereign data sharing. By coupling the knowlEdge cloud-to-edge continuum with concepts coming from IDSA[34], FIWARE[35], GAIA-X[36], etc., it will be possible to create a human-centered AI orchestration framework that can be adopted by many European industries and factories; this will extend current European manufacturing data space approaches [72] to include AI orchestration and human-centered concepts rather than only data with relevant usage policies and governance rules). Contributing to or leveraging these standards will ensure seamless interoperability with other systems and data spaces within the manufacturing ecosystem.

In conclusion, the market success of the knowlEdge platform depends on its compatibility and interoperability; these are critical factors that will determine its widespread adoption. The platform's resilience is further bolstered by its ability to cater to a diverse user base and diverse industries. Integrating robust feedback mechanisms ensures ongoing refinement, enabling it to adapt to evolving user needs. By allowing the incorporation of further emerging technologies and capitalizing on their diverse benefits, the knowlEdge platform promises continuous support and timely updates and anticipates future developments. This multifaceted approach not only secures the platform's immediate relevance but also lays the foundation for its long-term sustainability.

**Data Availability Statement:** Data is contained within the article.

## Notes

1    https://developer.nvidia.com/nvidia-triton-inference-server (accessed on 22 January 2024)
2    https://www.verta.ai/platform/model-registry (accessed on 22 January 2024)
3    https://aimodelplace.com/ (accessed on 22 January 2024)
4    https://www.modzy.com/marketplace (accessed on 22 January 2024)
5    https://beta.singularitynet.io (accessed on 22 January 2024)
6    https://gaia-x.eu/ (accessed on 22 January 2024)
7    https://dataspaces.info/common-european-data-spaces/ (accessed on 22 January 2024)
8    https://www.nextworks.it/en/products/symphony (accessed on 22 January 2024)
9    PMML provides a standard way to represent trained machine learning and statistical models, making it easier to exchange models between different platforms and tools.
10   ONNX stands for open neural network exchange. It is an open source format and ecosystem designed to facilitate interoperability between different deep learning frameworks and tools.
11   Docker SDK is a Python library for the Docker Engine API. It lets you do anything the Docker command does—run containers, manage containers, manage Swarms, etc.—but from within the Python app.
12   An Ontology is a formal, explicit specification of a shared conceptualization of a domain. It provides a structured representation of knowledge about a particular domain, defining the concepts, relationships, and properties that exist within that domain.
13   A case-based reasoning system is an artificial intelligence (AI) approach that solves new problems by reusing solutions from similar past cases. It is a problem-solving methodology that relies on the retrieval and adaptation of previous cases to address new situations.
14   https://hub.docker.com/_/docker (accessed on 22 January 2024)
15   https://prometheus.io/ (accessed on 22 January 2024)
16   https://grafana.com/oss/ (accessed on 22 January 2024)
17   https://github.com/google/cadvisor (accessed on 22 January 2024)
18   https://github.com/prometheus/node_exporter (accessed on 22 January 2024)
19   https://onnx.ai/ (accessed on 22 January 2024)
20   https://threejs.org (accessed on 22 January 2024)
21   LINKS Foundation, Via Pier Carlo Boggio 61, 10138 Torino, Italy.
22   European Commission, High-Level Expert Group on AI (HLEG): Ethics Guidelines for Trustworthy Artificial Intelligence, 2019. URL: http://doi.org/10.2759/346720 (accessed on 22 January 2024)
23   https://kubernetes.io/ (accessed on 22 January 2024)
24   https://cloudify.co/ (accessed on 22 January 2024)
25   https://docs.docker.com/engine/swarm/ (accessed on 22 January 2024)
26   https://aws.amazon.com/cloudformation/ (accessed on 22 January 2024)
27   https://cloud.google.com/deployment-manager/ (accessed on 22 January 2024)
28   https://documentation.mindsphere.io/resources/html/manage-my-sinumerik-edge-app-publishing/en-US/user-docu/industrialedge.html (accessed on 22 January 2024)
29   https://mall.industry.siemens.com/mall/en/WW/Catalog/Products/10348389 (accessed on 22 January 2024)
30   https://blogs.sap.com/tags/73554900100700002011/ (accessed on 22 January 2024)
31   https://www.sap.com/products/technology-platform/cloud-analytics.html (accessed on 22 January 2024)
32   https://azure.microsoft.com/en-us/products/iot-edge (accessed on 22 January 2024)
33   https://aws.amazon.com/edge/ (accessed on 22 January 2024)
34   https://internationaldataspaces.org/ (accessed on 22 January 2024)
35   https://www.fiware.org/ (accessed on 22 January 2024)
36   https://www.data-infrastructure.eu/GAIAX/Navigation/EN/Home/home.html (accessed on 22 January 2024)

# References

1. European Commission; Directorate-General for Research and Innovation; Müller, J. *Enabling Technologies for Industry 5.0: Results of a Workshop with Europe's Technology Leaders*; Publications Office of the European Union: Brussels, Belgium, 2020. [CrossRef]
2. European Commission; Directorate-General for Research and Innovation; Breque, M.; De Nul, L.; Petridis, A. *Industry 5.0: Towards a Sustainable, Human-Centric and Resilient European Industry*; Publications Office of the European Union: Brussels, Belgium, 2021. [CrossRef]
3. European Commission; Directorate-General for Research and Innovation; Renda, A.; Schwaag Serger, S.; Tataj, D.; Morlet, A.; Isaksson, D.; Martins, F.; Mir Roca, M.; Hidalgo, C.; et al. *Industry 5.0, a Transformative Vision for Europe: Governing Systemic Transformations towards a Sustainable Industry*; Publications Office of the European Union: Brussels, Belgium, 2022. [CrossRef]
4. ManuFUTURE High-level Group. *ManuFUTURE Strategic Research Agenda SRIA 2030. For a Competitive, Sustainable and Resilient European Manufacturing*; ManuFUTURE: Brussels, Belgium, 2019.
5. Westkämper, E. Manufacturing the Backbone of the European Economy. In *Towards the Re-Industrialization of Europe: A Concept for Manufacturing for 2030*; Springer: Berlin/Heidelberg, Germany, 2014; pp. 7–16. [CrossRef]
6. Siagian, H.; Tarigan, Z.J.H.; Jie, F. Supply Chain Integration Enables Resilience, Flexibility, and Innovation to Improve Business Performance in COVID-19 Era. *Sustainability* **2021**, *13*, 4669. [CrossRef]
7. Masoud Kamalahmadi, M.S.; Parast, M.M. The impact of flexibility and redundancy on improving supply chain resilience to disruptions. *Int. J. Prod. Res.* **2022**, *60*, 1992–2020. [CrossRef]
8. Kazancoglu, I.; Ozbiltekin-Pala, M.; Kumar Mangla, S.; Kazancoglu, Y.; Jabeen, F. Role of flexibility, agility and responsiveness for sustainable supply chain resilience during COVID-19. *J. Clean. Prod.* **2022**, *362*, 132431. . [CrossRef]
9. Moosavi, J.; Fathollahi-Fard, A.M.; Dulebenets, M.A. Supply chain disruption during the COVID-19 pandemic: Recognizing potential disruption management strategies. *Int. J. Disaster Risk Reduct.* **2022**, *75*, 102983. [CrossRef] [PubMed]
10. Grewal, R.; Tansuhaj, P. Building Organizational Capabilities for Managing Economic Crisis: The Role of Market Orientation and Strategic Flexibility. *J. Mark.* **2001**, *65*, 67–80. [CrossRef]
11. Davis, J.; Edgar, T.; Porter, J.; Bernaden, J.; Sarli, M. Smart manufacturing, manufacturing intelligence and demand-dynamic performance. *Comput. Chem. Eng.* **2012**, *47*, 145–156; [CrossRef]
12. Bethune, E.; Buhalis, D.; Miles, L. Real time response (RTR): Conceptualizing a smart systems approach to destination resilience. *J. Destin. Mark. Manag.* **2022**, *23*, 100687. [CrossRef]
13. Jovane, F.; Westkämper, E.; Williams, D. *The ManuFuture Road. Towards Competitive and Sustainable High-Adding-Value Manufacturing*; Springer: Berlin, Germany, 2009.
14. Carvalho, V.M.; Tahbaz-Salehi, A. Production Networks: A Primer. *Annu. Rev. Econ.* **2019**, *11*, 635–663. [CrossRef]
15. Esmaeilian, B.; Behdad, S.; Wang, B. The evolution and future of manufacturing: A review. *J. Manuf. Syst.* **2016**, *39*, 79–100. [CrossRef]
16. David R.S.; Vinit Parida, M.L.; Petrovic, A. Smart Factory Implementation and Process Innovation. *Res.-Technol. Manag.* **2018**, *61*, 22–31. [CrossRef]
17. Panetto, H.; Iung, B.; Ivanov, D.; Weichhart, G.; Wang, X. Challenges for the cyber-physical manufacturing enterprises of the future. *Annu. Rev. Control.* **2019**, *47*, 200–213. [CrossRef]
18. Bueno, A.; Godinho Filho, M.; Frank, A.G. Smart production planning and control in the Industry 4.0 context: A systematic literature review. *Comput. Ind. Eng.* **2020**, *149*, 106774. [CrossRef]
19. Herrmann, A.; Huber, F.; Braunstein, C. Market-driven product and service design: Bridging the gap between customer needs, quality management, and customer satisfaction. *Int. J. Prod. Econ.* **2000**, *66*, 77–96. . [CrossRef]
20. Benbarrad, T.; Salhaoui, M.; Kenitar, S.B.; Arioua, M. Intelligent Machine Vision Model for Defective Product Inspection Based on Machine Learning. *J. Sens. Actuator Netw.* **2021**, *10*, 7. [CrossRef]
21. Swanepoel, K.T. Decision support system: Real-time control of manufacturing processes. *J. Manuf. Technol. Manag.* **2004**, *15*, 68–75. [CrossRef]
22. Hossein Tehrani Nik Nejad, N.S.; Iwamura, K. Agent-based dynamic integrated process planning and scheduling in flexible manufacturing systems. *Int. J. Prod. Res.* **2011**, *49*, 1373–1389. [CrossRef]
23. Minguillon, F.E.; Lanza, G. Coupling of centralized and decentralized scheduling for robust production in agile production systems. *Procedia CIRP* **2019**, *79*, 385–390. [CrossRef]
24. Puchkova, A.; McFarlane, D.; Srinivasan, R.; Thorne, A. Resilient planning strategies to support disruption-tolerant production operations. *Int. J. Prod. Econ.* **2020**, *226*, 107614. [CrossRef]
25. Gunasekaran, A.; Yusuf, Y.Y. Agile manufacturing: A taxonomy of strategic and technological imperatives. *Int. J. Prod. Res.* **2002**, *40*, 1357–1385. [CrossRef]
26. Kagermann, H. Change through Digitization—Value Creation in the Age of Industry 4.0. In *Management of Permanent Change*; Albach, H., Meffert, H., Pinkwart, A., Reichwald, R., Eds.; Springer Fachmedien Wiesbaden: Wiesbaden, Germany, 2015; pp. 23–45. [CrossRef]
27. Romero, D.; Stahre, J.; Wuest, T.; Noran, O.; Bernus, P.; Berglund, Å.F.; Gorecky, D. Towards an Operator 4.0 Typology: A Human-Centric Perspective on the Fourth Industrial Revolution Technologies. In Proceedings of the International Conference on Computers & Industrial Engineering (CIE46), Tianjin, China, 29–31 October 2016; pp. 29–31.

28. Aldoseri, A.; Al-Khalifa, K.; Hamouda, A. A Roadmap for Integrating Automation with Process Optimization for AI-powered Digital Transformation. 2023 *preprints* [CrossRef]

29. Molina, A. Emerging Approaches for Enterprises and Human Integration towards Industry 5.0. In *Collaborative Networks in Digitalization and Society 5.0*; Camarinha-Matos, L.M., Boucher, X., Ortiz, A., Eds.; Springer Nature: Cham, Switzerland, 2023; pp. 353–364.

30. Shahrokni, A.; Söderberg, J. Beyond Information Silos Challenges in Integrating Industrial Model-based Data. In Proceedings of the 3rd Workshop on Scalable Model Driven Engineering, L'Aquila, Italy, 23 July 2015; Volume 1406. Available online: http://nbn-resolving.de/urn:nbn:de:0074-1406-4 (accessed on 22 January 2024).

31. Răileanu, S.; Anton, F.; Borangiu, T.; Anton, S.; Nicolae, M. A cloud-based manufacturing control system with data integration from multiple autonomous agents. *Comput. Ind.* **2018**, *102*, 50–61. [CrossRef]

32. Milojicic, D. The edge-to-cloud continuum. *Computer* **2020**, *53*, 16–25. [CrossRef]

33. Tuli, S.; Mirhakimi, F.; Pallewatta, S.; Zawad, S.; Casale, G.; Javadi, B.; Yan, F.; Buyya, R.; Jennings, N.R. AI augmented Edge and Fog computing: Trends and challenges. *J. Netw. Comput. Appl.* **2023**, *216*, 103648. [CrossRef]

34. Xie, Y.; Cruz, L.; Heck, P.; Rellermeyer, J.S. Systematic mapping study on the machine learning lifecycle. In Proceedings of the 2021 IEEE/ACM 1st Workshop on AI Engineering-Software Engineering for AI (WAIN), Madrid, Spain, 30–31 May 2021; pp. 70–73.

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

36. Pauli, T.; Fielt, E.; Matzner, M. Digital industrial platforms. *Bus. Inf. Syst. Eng.* **2021**, *63*, 181–190. [CrossRef]

37. Schermuly, L.; Schreieck, M.; Wiesche, M.; Krcmar, H. *Developing an Industrial IoT Platform—Trade-Off between Horizontal and Vertical Approaches*; In Proceedings of the 14. Internationale Tagung Wirtschaftsinformatik, Siegen, Germany, 24–27 February 2019.

38. Sisinni, E.; Saifullah, A.; Han, S.; Jennehag, U.; Gidlund, M. Industrial internet of things: Challenges, opportunities, and directions. *IEEE Trans. Ind. Inform.* **2018**, *14*, 4724–4734. [CrossRef]

39. Raj, E.; Buffoni, D.; Westerlund, M.; Ahola, K. Edge MLOps: An Automation Framework for AIoT Applications. In Proceedings of the 2021 IEEE International Conference on Cloud Engineering (IC2E), Virtual, 4–8 October 2021; pp. 191–200. [CrossRef]

40. Soto, J.A.C.; Jentsch, M.; Preuveneers, D.; Ilie-Zudor, E. CEML: Mixing and Moving Complex Event Processing and Machine Learning to the Edge of the Network for IoT Applications. In Proceedings of the 6th International Conference on the Internet of Things, Stuttgart, Germany, 7–9 November 2016; pp. 103–110. [CrossRef]

41. Urbanowicz, R.J.; Moore, J.H. ExSTraCS 2.0: Description and evaluation of a scalable learning classifier system. *Evol. Intell.* **2015**, *8*, 89–116. [CrossRef]

42. Nguyen, A.T.; Kharosekar, A.; Krishnan, S.; Krishnan, S.; Tate, E.; Wallace, B.C.; Lease, M. Believe It or Not: Designing a Human-AI Partnership for Mixed-Initiative Fact-Checking. In Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology, Berlin, Germany, 14–17 October 2018; pp. 189–199. [CrossRef]

43. Khadpe, P.; Krishna, R.; Fei-Fei, L.; Hancock, J.T.; Bernstein, M.S. Conceptual Metaphors Impact Perceptions of Human-AI Collaboration. *Proc. ACM Hum.-Comput. Interact.* **2020**, *4*, 1–26. [CrossRef]

44. Li, T.; Vorvoreanu, M.; DeBellis, D.; Amershi, S. Assessing Human-AI Interaction Early through Factorial Surveys: A Study on the Guidelines for Human-AI Interaction. *ACM Trans. Comput.-Hum. Interact.* **2022**, *30*, 1–45. [CrossRef]

45. Fan, M.; Yang, X.; Yu, T.; Liao, Q.V.; Zhao, J. Human-AI Collaboration for UX Evaluation: Effects of Explanation and Synchronization. *Proc. ACM Hum.-Comput. Interact.* **2022**, *6*, 1–32. [CrossRef]

46. Mucha, H.; Robert, S.; Breitschwerdt, R.; Fellmann, M. Interfaces for Explanations in Human-AI Interaction: Proposing a Design Evaluation Approach. In Proceedings of the Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems, New York, NY, USA, 8–13 May 2021. [CrossRef]

47. Álvarez-Napagao, S.; Ashmore, B.; Barroso, M.; Barrué, C.; Beecks, C.; Berns, F.; Bosi, I.; Chala, S.A.; Ciulli, N.; Garcia-Gasulla, M.; et al. knowlEdge Project -Concept, Methodology and Innovations for Artificial Intelligence in Industry 4.0. In Proceedings of the 19th IEEE International Conference on Industrial Informatics, INDIN 2021, Palma de Mallorca, Spain, 21–23 July 2021; pp. 1–7. [CrossRef]

48. Wajid, U.; Nizamis, A.; Anaya, V. Towards Industry 5.0—A Trustworthy AI Framework for Digital Manufacturing with Humans in Control. In Proceedings of the Workshop of I-ESA'22, Valencia, Spain, 23–24 March 2022; Volume 1613, p. 0073. Available online: http://nbn-resolving.de/urn:nbn:de:0074-3214-0 (accessed on 22 January 2024).

49. Bengio, Y.; Courville, A.; Vincent, P. Representation Learning: A Review and New Perspectives. 2014. Available online: http://xxx.lanl.gov/abs/1206.5538 (accessed on 22 January 2024).

50. Grinberg, M. *Flask Web Development: Developing Web Applications with Python*; O'Reilly Media, Inc.: Sebastopol, CA, USA, 2018.

51. Merkel, D. Docker: Lightweight linux containers for consistent development and deployment. *Linux J.* **2014**, *239*, 2.

52. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.

53. Mihai, A. Intelligent Decision Support System for Machine Learning Algorithms Recommendation. Master's Thesis, Universitat Politècnica de Catalunya (UPC), Barcelona, Spain, January/February 2017. Available online: http://hdl.handle.net/2117/102363 (accessed on 22 January 2024).

54. Gibert, K.; Sànchez-Marrè, M.; Codina, V. Choosing the Right Data Mining Technique: Classification of Methods and Intelligent Recommendation. In Proceedings of the 5th International Congress on Environmental Modelling and Software (iEMSs'2010), Ottawa, ON, Canada, 5–8 July 2010; Volume 3, pp. 1940–1947. Available online: https://scholarsarchive.byu.edu/iemssconference/2010/all/453/ (accessed on 22 January 2024).

55. Arinez, J.F.; Chang, Q.; Gao, R.X.; Xu, C.; Zhang, J. Artificial intelligence in advanced manufacturing: Current status and future outlook. *J. Manuf. Sci. Eng.* **2020**, *142*, 110804. [CrossRef]

56. Gunasekaran, A.; Yusuf, Y.Y.; Adeleye, E.O.; Papadopoulos, T.; Kovvuri, D.; Geyi, D.G. Agile manufacturing: An evolutionary review of practices. *Int. J. Prod. Res.* **2019**, *57*, 5154–5174. [CrossRef]

57. Van Rossum, G.; Drake, F.L. *Python 3 Reference Manual*; CreateSpace: Scotts Valley, CA, USA, 2009.

58. Banker, K.; Garrett, D.; Bakkum, P.; Verch, S. *MongoDB in Action: Covers MongoDB*, version 3.0; Simon and Schuster: New York, NY, USA, 2016.

59. White, T. *Hadoop: The Definitive Guide*; O'Reilly Media, Inc.: Sebastopol, CA, USA, 2012.

60. Guazzelli, A.; Zeller, M.; Lin, W.C.; Williams, G. PMML: An open standard for sharing models. *R J.* **2009**, *1*, 60. [CrossRef]

61. Publio, G.C.; Esteves, D.; Ławrynowicz, A.; Panov, P.; Soldatova, L.; Soru, T.; Vanschoren, J.; Zafar, H. ML-schema: Exposing the semantics of machine learning with schemas and ontologies. *arXiv* **2018**, arXiv:1807.05351.

62. Kritzinger, W.; Karner, M.; Traar, G.; Henjes, J.; Sihn, W. Digital Twin in manufacturing: A categorical literature review and classification. *Ifac-PapersOnline* **2018**, *51*, 1016–1022. [CrossRef]

63. Moiceanu, G.; Paraschiv, G. Digital twin and smart manufacturing in industries: A bibliometric analysis with a focus on industry 4.0. *Sensors* **2022**, *22*, 1388. [CrossRef] [PubMed]

64. Holopainen, M.; Saunila, M.; Rantala, T.; Ukko, J. Digital twins' implications for innovation. *Technol. Anal. Strateg. Manag.* **2022**, 1–13. [CrossRef]

65. Xiong, Y.; Sun, Y.; Xing, L.; Huang, Y. Extend Cloud to Edge with KubeEdge. In Proceedings of the 2018 IEEE/ACM Symposium on Edge Computing (SEC), Seattle, WA, USA, 25–27 October 2018; pp. 373–377. [CrossRef]

66. Ullah, A.; Dagdeviren, H.; Ariyattu, R.C.; DesLauriers, J.; Kiss, T.; Bowden, J. Micado-edge: Towards an application-level orchestrator for the cloud-to-edge computing continuum. *J. Grid Comput.* **2021**, *19*, 47. [CrossRef]

67. Rousopoulou, V.; Vafeiadis, T.; Nizamis, A.; Iakovidis, I.; Samaras, L.; Kirtsoglou, A.; Georgiadis, K.; Ioannidis, D.; Tzovaras, D. Cognitive analytics platform with AI solutions for anomaly detection. *Comput. Ind.* **2022**, *134*, 103555. [CrossRef]

68. Akundi, A.; Euresti, D.; Luna, S.; Ankobiah, W.; Lopes, A.; Edinbarough, I. State of Industry 5.0—Analysis and Identification of Current Research Trends. *Appl. Syst. Innov.* **2022**, *5*, 27. [CrossRef]

69. Leng, J.; Sha, W.; Wang, B.; Zheng, P.; Zhuang, C.; Liu, Q.; Wuest, T.; Mourtzis, D.; Wang, L. Industry 5.0: Prospect and retrospect. *J. Manuf. Syst.* **2022**, *65*, 279–295. [CrossRef]

70. Maddikunta, P.K.R.; Pham, Q.V.; B, P.; Deepa, N.; Dev, K.; Gadekallu, T.R.; Ruby, R.; Liyanage, M. Industry 5.0: A survey on enabling technologies and potential applications. *J. Ind. Inf. Integr.* **2022**, *26*, 100257. [CrossRef]

71. ConnectedFactories 2. D2.6 Pathways Cross-Fertilisation with Digital Technologies—Second Iteration. In *Deliverable of Connected Factories 2 Consortium*; Connected Factories 2 Consortium, 2022. Available online: www.connectedfactories.eu (accessed on 22 January 2024).

72. Mertens, C.; Alonso, J.; Lázaro, O.; Palansuriya, C.; Böge, G.; Nizamis, A.; Rousopoulou, V.; Ioannidis, D.; Tzovaras, D.; Touma, R.; et al. A Framework for Big Data Sovereignty: The European Industrial Data Space (EIDS). In *Data Spaces: Design, Deployment and Future Directions*; Springer International Publishing: Cham, Switzerland, 2022; pp. 201–226.