AIDOaRt: AI-augmented Automation for DevOps, a Model-based Framework for Continuous Development in Cyber-Physical Systems

Romina Eramo, Vittoriano Muttillo, Luca Berardinelli, Hugo Bruneliere, Abel Gomez, Alessandra Bagnato, Andrey Sadovykh, Antonio Cicchetti

AIDOaRt: AI-augmented Automation for DevOps, a Model-based Framework for Continuous Development in Cyber-Physical Systems

Romina Eramo, Vittoriano Muttillo
University of L'Aquila
L'Aquila, Italy
{romina.eramo, vittoriano.muttillo}@univaq.it

Luca Berardinelli
Johannes Kepler University
Linz, Austria
luca.berardinelli@jku.at

Hugo Bruneliere
IMT Atlantique, LS2N (UMR CNRS 6004)
Nantes, France
hugo.bruneliere@imt-atlantique.fr

Abel Gomez
Universitat Oberta de Catalunya
Barcelona, Spain
agomezlla@uoc.edu

Alessandra Bagnato, Andrey Sadovykh
Softeam
Paris, France
{alessandra.bagnato,andrey.sadovykh}@softeam.fr

Antonio Cicchetti
Mälardalen University
Västerås, Sweden
antonio.cicchetti@mdh.se

Abstract—With the emergence of Cyber-Physical Systems (CPS), the increasing complexity in development and operation demands for an efficient engineering process. In the recent years DevOps promotes closer continuous integration of system development and its operational deployment perspectives. In this context, the use of Artificial Intelligence (AI) is beneficial to improve the system design and integration activities, however, it is still limited despite its high potential. AIDOaRT is a 3 years long H2020-ECSEL European project involving 32 organizations, grouped in clusters from 7 different countries, focusing on AI-augmented automation supporting modelling, coding, testing, monitoring and continuous development of Cyber-Physical Systems (CPS). The project proposes to apply Model-Driven Engineering (MDE) principles and techniques to provide a framework offering proper AI-enhanced methods and related tooling for building trustable CPSs. The framework is intended to work within the DevOps practices combining software development and information technology (IT) operations. In this regard, the project points at enabling AI for IT operations (AIOps) to automate decision making process and complete system development tasks. This paper presents an overview of the project with the aim to discuss context, objectives and the proposed approach.

Index Terms—Continuous System Engineering, DevOps, AIOPS, Software engineering, Scientific computing, Simulation and Modelling Tools, Cyber-Physical Systems, Artificial Intelligence

I. Introduction

Nowadays, modern systems in the domains of Industry 4.0, healthcare, autonomously driving cars or smart grids are examples of highly communicating systems where software enables increasingly advanced functionality. The growing complexity of these Cyber-Physical Systems (CPS) [1] poses several challenges throughout all system design, development and analysis phases, but also during their deployment, actual usage and future maintenance.

In this context, Model Driven Engineering (MDE) [2], [3] has gained popularity in software development projects as it allows raising the level of abstraction and thus improving the ability to handle complexity of systems. The usage of models, as purposeful abstractions of systems and of their environment, is a pillar enabling technology in current and future CPS engineering platforms [1].

With the advent of DevOps principles [4], [5], the engineering of CPS would benefit from supporting a continuous development involving a smooth continuum of information from design to runtime, and vice versa. Moreover, many leading companies have started envisioning the productivity boost of tomorrow to be brought by the application of full-blown Artificial Intelligence (AI) principles and techniques [6]. By means of AI for IT operations (AIOps) [7], we can rethink the DevOps pipeline through continuous monitoring, alerting and remediation in a secure and reliable way. However, to the best of our knowledge, there has not been any general and reusable AI-augmented approach intending to support full continuous software and system engineering processes for CPSs.

The AIDOaRt project [8] aims at providing a model-based framework to more efficiently support the continuous software and system engineering of CPSs via AI-augmentation. Complementary to the support for already existing systems of any kind, AIDOaRt also aims at improving the continuous development of new modern CPSs. AIDOaRt is a 3 years long H2020-ECSEL European project involving 32 organizations, grouped in clusters from 7 different countries. The starting date of the project was April 1, 2021. In practice, AIDOaRt aims at enhancing the DevOps tool chain by employing AI techniques, in particular Machine Learning (ML), in multiple aspects of the system engineering process such as design and modeling quality, source code performance, testing coverage, predictive monitoring. DevOps teams may use the AIDOaRt framework to analyze event streams (for real-time and historical data) together with the design information e.g. available as suitably interwoven models, in order to extract meaningful insights for system continuous improvement through development and

faster deployments, to foster better collaboration, and to reduce downtime with proactive detection approaches. We expect an industrial uptake of AIDOaRt technologies on the development of complex systems that scales up to real systems' demand with relevance for all critical applications.

The rest of the paper is structured as follows. Section II outlines the main project's objectives and introduces the concepts the AIDOaRt project envisions to builds up on; Section III presents in more details the AIDOaRt conceptual architecture and overall approach; Section IV provides an overview of the AIDOaRt industrial use cases; Section V summarizes the current work plan and expected results. Finally, Section VI concludes the paper.

II. AIDOaRt Objectives and Concepts

This section introduces the main AIDOaRt objectives and the main elements of the underlying conceptual framework to accomplish the envisioned AIDOaRt goals.

A. Objectives

AIDOaRt aims at providing a holistic approach for the continuous systems engineering [9] that:

1) Provides a core model-based framework to support the continuous systems engineering process of CPSs by benefiting from AI-augmented automation Concerning the MDE aspect, we take advantage of (parts of) existing propositions, approaches or tools resulting from, Megamart2 [10]), where many AIDOaRt partners were involved;
2) Enhances available DevOps tool chain by integrating the use of AI techniques and notably Machine Learning (ML) in multiple aspects of the system development process in particular to support requirements, monitoring, modeling, coding and testing activities;
3) Supports the collection, representation and traceability of runtime data, such as logs, events and metrics, and software models (Observe), assists in the analysis of both historical and real time data in combination with design information (Analyze) and supports the automation of tasks of the DevOps pipeline according to the results of previous analysis (Automate).We call these activities Observe, Analyze and Automate later in the paper.

Therefore, in order to fully realize this potential, the AIDOaRt consortium aims at accomplishing the following mission: creating a framework incorporating methods and tools for continuous software and system engineering and validation, leveraging the advantages of AI techniques. As the result, we expect a significant improvement in productivity, quality, and predictability of CPSs and, more in general, of large and complex industrial systems.

B. AIDOaRt concepts

The overall idea of AIDOaRt is to efficiently support the system engineering life cycle, from requirements to testing and deployment, including software design and coding. In particular, AIDOaRt focuses on supporting the continuous development of Cyber-Physical Systems (CPSs) via AI-augmented

automation. To this intent, AIDOaRt proposes the use of MDE principles and techniques to provide a model-based framework offering proper methods and related tooling. This framework will enable the observation and analysis of collected data from both runtime and design time in order to provide dedicated AI-augmented solutions, that will then be validated in concrete industrial cases involving complex CPSs, as introduced later in Section IV.

Model Driven Engineering (MDE). Over the last two decades, MDE has become increasingly popular in software development projects. Indeed, it allows raising the level of abstraction and thus improving the ability to handle complex systems such as CPSs [1] The usage of models, as purposeful abstractions of systems and of their environment, is also of increasing importance for industrial applications, leading to new challenges e.g., digital twining [11]. The first generation MDE tools mostly focused on generating code from high-level platform-independent abstract descriptions. Since then, the target scope of MDE has broadened and now addresses model-based testing, verification, measurement, tool interoperability or software evolution, among many difficult challenges in software engineering. In this project, MDE will contribute by (i) providing better abstraction principles and techniques, (ii) facilitating activities automation, and (iii) supporting technology integration among all the covered design and development activities. In AIDOaRt, the heterogeneous data resulting from different engineering activities will be converted and then manipulated as MDE artefacts, contributing to make explicit the implicit modeling as performed in other project areas. By switching to model driven approaches, businesses are promised to reap benefits through increased productivity and software quality. MDE is the software engineering paradigm we plan to rely on in AIDOaRt in order to design and develop our framework.

Artificial Intelligence and Machine Leaning (AI/ML). The use of AI in general, and ML in particular, is another key aspect of the AIDOaRt project. The dissemination of such principles and techniques in a regulated industry can rapidly enable systems to decide and act in a more and more automated manner, sometimes even without direct human control [12]. As a consequence, this demands for a responsible approach to ensure a safe and beneficial use of AI technologies. This approach has to consider both the implications of (co)decision making by machines and related ethical issues cf. the definition of AI legal status [13]. Within AIDOaRt, we aim at 1) ensuring that systems are designed responsibly and 2) contributing to our trust on their behavior. This notably requires supporting both **explainability** [14] (i.e., being able to explain and justify decisions), and **accountability** [15] (i.e., justifying, based on given explanations, decisions and actions taken by stakeholders involved in the CPS engineering process).

DevOps. AIDOaRt also aims at contributing to the adoption of DevOps [5], [16], [17]. DevOps is defined as a softwar engineernig paradigm focused on software delivery through enabling continuous feedback, quick response to changes and

using automated delivery pipelines resulting in reduced cycle time. In particular, within DevOps, tools and methods are focused on administration and automation processes. AIDOaRt proposes an overall model-based approach combining existing techniques to be enhanced

Artificial Intelligence for IT Operations (AIOps). The term AIOps was coined by Gartner [6], [18] to characterize solutions where DevOps challenges are addressed with the help of AI/ML techniques. A fundamental challenge notably resides in integrating AIOps [7] in the process of enhancing the DevOps pipeline, for example through continuous alert and insights from data used for continuous deployment and operations management. AIOps not only uses data science and computational techniques to automate common and routine operational tasks, it also digests metrics and uses inference models to extract actionable insights from data. We believe AI is a relevant complementary mean, along with MDE, to achieve AIDOaRt goals, providing a more efficient and flexible support to the continuous development of CPSs.

Data. Within AIDOaRt, data represents a key aspect as far as the combination of MDE and AI is concerned. From a MDE perspective, everything is a model including all kinds of data (e.g., monitored logs and execution traces). In the project, data will be collected both at runtime and design time; furthermore, relevant traceability information between these two levels will be computed and maintained. Once available, all this data needs to be handled by relying on a combination of MDE and AI techniques. In this respect, all relevant data in the context of AIDOaRt will be treated as models, enabling share and (re)use in different contexts and by different technical solutions of the framework (including AI ones) [19]. Moreover, this will support interoperability between different data formats.

Fig. 1 provides a conceptual architecture of the AIDOaRt global approach and emphasizes on its key principles and concepts. The overall AIDOaRt infrastructure consumes different kinds of data, including notably runtime data (e.g., IT monitoring, log events, etc.), along with design data produced during the software development process (e.g., software models, design documentation, traceability information, source code, etc.). All collected data will be processed, translated in an internal model-based representation, and collected in a shared repository (*Data Collection & Model-based Representation* component of Fig. 1. The *Model-based Core Infrastructure* component is intended to support the DevOps practices efficiently combining software development and IT operations.

Moreover, AIDOaRt aims at enhancing DevOps tool chains (see [20] for a list of DevOps tools) by employing AI and ML techniques in multiple aspects of the system development process (e.g., requirements, monitoring, modeling, coding, and testing), and made them available as a *AI-augmented Toolkit*. In a AIOps [6]-enabled context, the toolkit should support i) the monitoring of runtime data (such as logs, events and metrics [21]) and software data and their traceability (*Observe*), ii) the analysis of both historical and real time data (*Analyze*), and iii) the automation of development and operation activities (*Automate*). Accountability [15] and
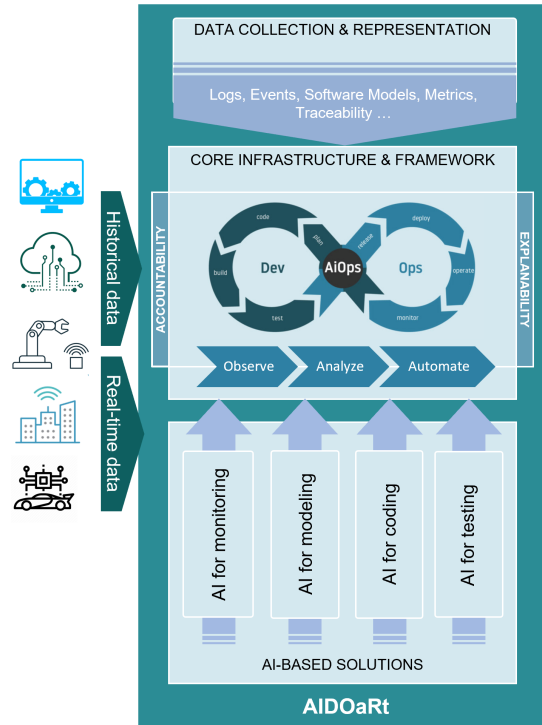

Figure 1. AIDOaRt Conceptual Architecture.

explainability [14] capabilities should be managed as cross-cutting concerns throughout the DevOps engineering process and tool chain. Accountability and explainability consuming available design-time and runtime data, which, according to MDE principles, should be made available to stakeholders as design-time and runtime models, respectively.

AIDOaRt aims at both extending existing techniques as well as introducing novel solutions. Notably, the state of the art of requirements engineering, monitoring, and testing, already include mechanisms supporting/leveraging data analysis [22]–[24]. Moreover, search-based techniques have been investigated as a way to automate MDE related activities such as language engineering, model transformation, model versioning [25]. Nonetheless, especially when dealing with mission critical systems, the automated generation of artefacts raises verification and validation issues, e.g. for certification purposes [26].

III. THE AIDOaRt APPROACH

The main innovation in AIDOaRt consists in bringing together DevOps with advanced MDE and AI principles and techniques. The AIDOaRt approach in Fig. 1 is detailed by an activity-like diagram in Fig. 2. Operational steps (blank rounded boxes) and input/output data/artifacts (blank squared boxes) are interconnected in a continuous flow of steps and corresponding input/output data/artifacts. A hierarchy of grey round boxes further aggregate steps and data/artifacts flowing in the envisioned AIDOaRt components (data collection and representation, core framework and infrastructure, AI-
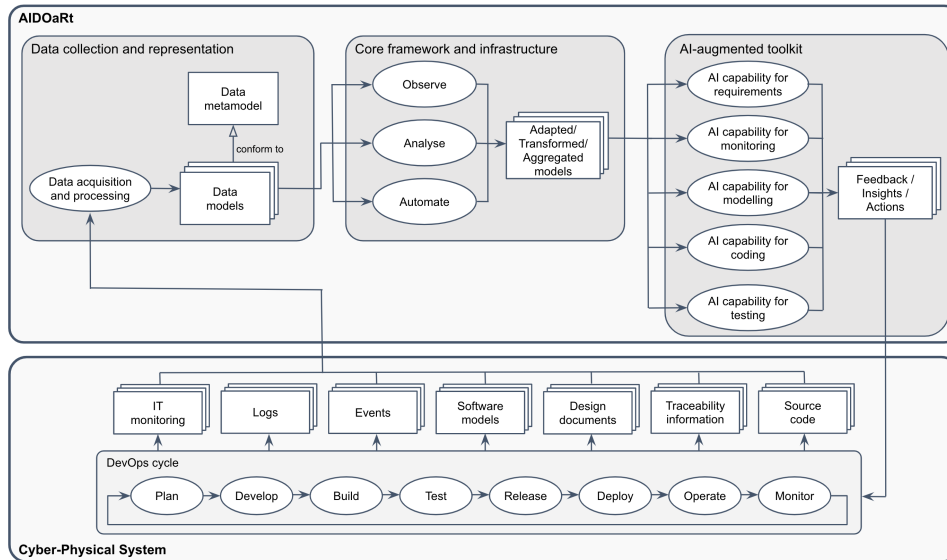
Figure 2. AIDOaRt Approach.

augmented toolkit) and typical phases of a DevOps cycle, from plan to monitor, applied to the given CPS.

During the development, several data/artifacts are produced, both at runtime and design time. Note that, we do not distinguish the nature of this data/artifacts, they can be models, text, databases, etc. We consider these as the dataset that the AIDOaRt *Data collection and representation* component acquires, elaborates and transforms into *Data models* that conform to a common *Data metamodel*. Such Data models are given as inputs to the *Core framework and infrastructure* component that provides the core model-based capabilities and services to support *Observe*, *Analyze* and *Automate* activities. The *Core framework and infrastructure* specifies and provides common interfaces/APIs to access these services, as well as some concrete tools implementing these APIs.

Finally, the AIOps operations provided by the *AI-augmented toolkit* component, powered by the AIDOaRtcore framework and infrastructure features, will support some typical engineering activities targeted in AIDOaRt, namely requirements, monitoring, modeling, coding, and testing ones. The output of these operations consists in feedback, insights or actions to be exploited in specific phases of the DevOps cycle.

In order to realize these contributions, we notably plan to base our work on previous initiatives from different past and present European projects [10], [27]. The objective is to i) reuse (parts of) the existing approaches or prototypes whenever relevant and ii) both complement and integrate them with new solutions designed and developed in AIDOaRt.

A. AIDOaRt Data Collection and Representation

The AIDOaRt *Data Collection and Representation* tooling will be designed and developed in order to support the management of the collected data that represents the initial input required by the AIDOaRt framework. The combination of real-time and historical data, along with data coming from system design e.g.

specifications, models, will allow the AIDOaRt framework to predict certain CPS properties and to help in making relevant decisions during the CPS engineering process. A successful framework for AIOps solutions has to be able to collect data from an entire multi-vendor and multi-domain environment, including network and storage solutions, containers, and cloud ifrastructure. As already introduced earlier, data will be collected both at runtime with IT monitoring, events logging etc. and at design time from software models, design documentation, traceability information, source code, etc. Thus, we will develop methods and supporting tools to acquire, correlate and provide access to a broad range of historical and real-time data types. To this end, AIDOaRt aims at reusing existing data management solutions (i.e., data platforms that permit real-time processing of streaming IT data like Hadoop 2.0, Elastic Stack, or some Apache technologies) that will be integrated with the AIDOaRt methods and tools. In this respect, the value of the AIOps solutions lies in their ability to ingest and correlate data across the environment. Runtime and design time data collected by the AIDOaRt framework will preserve relevant traceability information among them, notably in order to allow providing a proper and reusable feedback from runtime to design time.

B. AIDOaRt Core Infrastructure and Framework

The AIDOaRt *Core Infrastructure and Framework* will be designed and developed in order to support the core capabilities to be provided in the context of the project. The AIDOaRt infrastructure will provide the base platform of AIDOaRt, which consists of computational, networking, and storage services to be provided in order to support the targeted engineering activities (i.e., requirements, monitoring, modeling, coding, and testing). The infrastructure will expose the AIDOaRt internal data models as obtained in the previous step, providing the model-based representation for every kind of data needed

in the framework. Thus, the end-users will invoke an extensible set of AIDOaRt services that, built on top infrastructural services, support the realisation of the engineering activities and specific analysis e.g. for explainability or accountability, according to the global AIDOaRt architecture.

In the AIDOaRt context, explainability is considered as the capability, within a given decision/engineering process, to generate explanations to stakeholders involved in the decision process of the concerned CPS e.g., modelers, programmers, testers, IT experts, as well as to recommend actions to be taken to improve the system life cycle. Following MDE principles, such an explanation, can be itself expressed as a model [12]. It must be noted that, in a highly regulated environment, ML or AI techniques in general are unable to fully make decisions for obliged entities e.g., transparency, audit trails and a risk-based approach are critical to compliance programs. In this context, explainability is the key feature that emphasises the importance of the human role in the decision making process i.e. all decisions must be verified to ensure that they are correct and that the reached conclusion is "explainable". Moreover, we recognize that comprehensive risk analysis and special requirements for data management are indispensable foundations for the use of AIs in regulated areas. Such aspects will be addressed from the first phases of the project e.g. definition of requirements for information management, definition of internal representation for entities of a domain that require to be commonly understood. The AIDOaRt Core Infrastructure and Framework will provide all the means that will be used in the AI-augmented tools to perform development tasks i.e. Observe, Analyze, and Automate - as described in the next section.

C. AIDOaRt AI-augmented Toolkit

In order to extend the AIDOaRt *Core Infrastructure and Framework*, according to the needs of various kinds of CPSs under development, complementary AI-augmented capabilities have to be provided and then reused accordingly. To this intent, the AIDOaRt *AI-augmented Toolkit* will be designed and developed in the context of the project. They offer additional capabilities related to different CPS development tasks[1]. The AIDOaRt *AI-augmented toolkit* will provide specific AI support to multiple phases of the system development process, as detailed in the following.

AI for requirements. We will develop different AI-based tools in order to cope with different operations. AI methods, in particular ML ones, will be used to support requirement elicitation, recommendation, and consistency checking at early design stage. To help the elicitation, we plan to take into account historical requirement documents and exploit AI/ML for enhancing analysis and supporting activities at later stages of the development process. AI techniques will also be used to recommend (types of) requirements considering the concerned

---

[1]Note that some important tasks such as deployment are not explicitly mentioned here. However, they will be facilitated by the general automation capabilities we plan to provide via the AIDOaRt Core Infrastructure and its AI-augmented Toolkit.

application domain and its related standards and guidelines. Tools based on AI, in particular on Automated Reasoning (AR) techniques will be developed in order to automatically check the fulfillment of a given standard/guideline. In this context, Knowledge Representation (KR) techniques will also be considered in order to leverage a formal background that will allow to obtain explainable results from the AI-based tools. Another important aspect that will be investigated is the consistency checking of the requirements. During the project will be developed AI-based tools using AR techniques, in order to return an explainable and provable answer in the case of (un)consistency of the considered pool of requirements or technical specifications. Finally, we will also design and develop AI-based tools to support the writing and management of requirements. In this case, we will be developed AI/ML approaches leveraging on natural language processing (NLP), in particular for the translation from natural language to the formal languages needed for the requirement analysis e.g. checking the requirement formulation against a guideline.

AI for monitoring. We will focus on AI-based tools and techniques in order to deal with run-time properties verification, system fault detection and forecasting for predictive maintenance. These tasks will be accomplished by means of AI/ML algorithms that, considering the result of trace analysis and related historical data, will allow to detect both performance and system errors.

AI for modeling. AI-based tools and techniques will be developed to support decision making, strategy planning and recommendations during the modeling activities. In particular, there will be a developed AI-based assistant that guides the engineer in the design of complex mixed criticality systems, by suggesting the appropriated attributes that would enable feasible deployments according to previously defined multiple optimization criteria and to recommend additional modeling concepts relevant to the model-in-construction. AI/ML techniques will be also developed to support the (semi-)automated creation and/or update of large and complex model views involving heterogeneous (design time and runtime) models. Several AI-based techniques will be also developed to support the refinement and the verification of the model. In particular, ML and AR techniques will be developed for design consistency checking, automated properties/specifications verification and for anomaly detection and root cause determination in temporal behavior, in particular when related to safety-critical applications.

AI for coding. Notably, from a MDE perspective, code can be considered an executable textual model. Nevertheless, AI-based tools for coding will be also developed during the project, in order to help the user providing users with no or very superficial programming experience to create programs based on ideograms and similar. In this case, AI/ML techniques will be mainly employed to learn code generation patterns from examples.

AI for testing. Finally, we will develop AI-based tools and techniques for testing at different levels. AI/ML techniques will be proposed to gather, examine, and observe data models

in order to generate reliable unit tests. Pattern recognition techniques will also be developed to identify high quality test suites e.g. set of tests with more failure probabilities.

D. AIDOaRt Integration

In order to apply and evaluate in practice the AIDOaRt *Core Infrastructure and Framework* in combination with the AIDOaRt *AI-augmented Toolkit*, an integration approach will be specified and developed. The proposed integration will notably be based on the use of standards for sharing the various underlying models/data and for allowing to combine efficiently the different tools resulting from AIDOaRt technical components (described in Sections III-A III-B and III-C). The use of MDE artefacts and methodologies does not limit the specific integration architecture to be used. On the contrary, the use of model-based representations of the data provided a common understanding no matter how the data is exposed and consumed e.g. data lakes, APIs, etc. For instance, existing APIs can be refactored e.g. extended, reduced, combined or split, by discovering and linking the models behind each individual API. On the one hand, thanks to this flexibility, the AIDOaRt integration approach aims at creating a global API exposing a unified data model by merging the data models of the initial APIs, as defined in the *Core framework* and exploited in the *AI-augmented toolkit*. On the other hand, AIDOaRt will be able to adapt the global API to the specific use cases, based on data and processing capabilities required for performing their AI-related operations and workflows.

As discussed above, the global integration architecture will be defined during the early phases of the project in order to define constraints and requirements. In order to tame the risk of a complex technical integration due to an expected technological heterogeneity, the integration will explicitly take into account customizable engineering processes but all sharing common MDE principles and practices (e.g., metamodels, models and transformations). For this purpose, technological interoperability will be enabled through the tight collaboration between academy, technology providers, and industrial partners in the project consortium.

IV. INDUSTRIAL USE CASES

AIDOaRt brings together prominent tool vendors and research organisations with state-of-the-art methods and tools that will be validated in highly relevant European industry case studies. The end users from the space, naval, railway, smart grid, smart warehouse and telecom industry domains will drive the project by providing real-world requirements and case studies as well as by validating and endorsing the AIDOaRt research and technical results.

The AIDOaRt Core Infrastructure and Framework will be applied in different industrial use cases. The variety of use cases will allow to validate the different AI-augmented extensions of the AIDOaRt Toolkit, i.e. requirements management, monitoring, modeling, coding and testing . In what follows, we provide a short summary of these use cases.

A. Abinsula: Safety critical systems in the automotive domain using disruption technology

The main goal of this use case is to propose an approach and apply it in an interesting case as for example the virtual mirrors in cars. Such a case would require a complete change of cars design, nonetheless having all the rear images from cameras opens a plethora of possibilities (data fusion, correlation of different images from different points of view), and with them also possible hazards in the car, not dangerous as a failure of the brakes but still potentially very relevant issues. The final result will be a working PoC for virtual mirrors in cars; to develop this system a new and comprehensive methodology for developing automotive applications using AI e ML is necessary.

B. AVL - AI supported Digital Twin Synthesis supporting secure vehicle development and testing for novel propulsion systems

The AVL use case consists of several main tasks along AI-augmented methods for vehicle modelling (design phase), development and test (including verifying testing equipment and testing methods) and vehicle operation (including data acquisition). DevOps loops are essential and will be performed with higher frequency between development/test and operation, and with lower frequency between operation and design.

C. Bombardier Transportation: DevOps for Railway Propulsion System Design

Bombardier Transportation is one of the largest manufacturers of railway equipment and a global leader in electric propulsion and auxiliary systems for railway vehicles. By applying the DevOps philosophy to the propulsion system development process, along with advanced physical modelling techniques and innovative uses of AI/ML for automation, it is expected that the project outcomes will have a significant impact in improving time to market, quality, investment and life cycle cost.

D. CAMEA: AI for Traffic Monitoring Systems

In CAMEA, a family of traffic monitoring systems is being developed. These systems are video-based (reading licence plates), radar-based (detection of objects and measuring their speed), using other sensors or a combination of the mentioned ones. Improving the power-aware development process of video-based systems, radar-based systems, or their fusion using AI can help tuning specific parameters and configuration of algorithms in a much faster and sophisticated way or finding better solutions.

E. ClearSy: Machine learning in interactive proving

The Atelier B and its interactive industrial-grade prover developed by ClearSy are used in the development of many safety critical systems in the railway domain and allows for the operational use of the B Method to develop defect-free proven software (formal software) [28]. Machine learning can help engineers in some aspects of the B development, being those interactive proving or adapting proof to change of hypothesis.
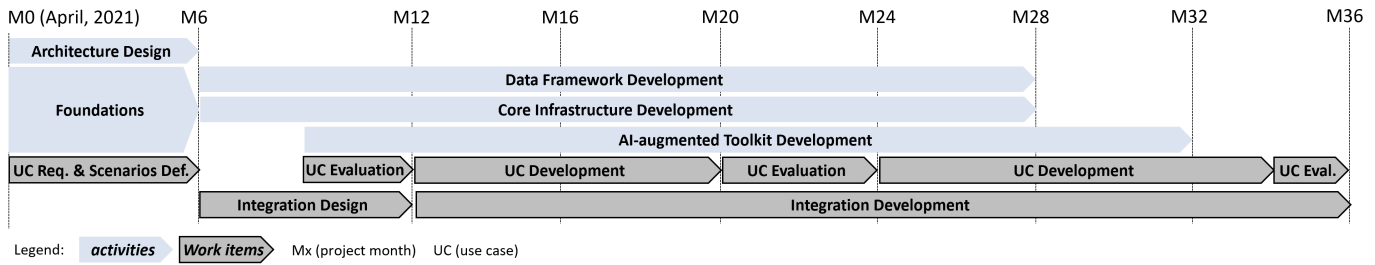
Figure 3. AIDOaRt timeline.

F. HI Iberia: AI DevOps in the restaurants business

HIPPOS [29] is a Point of Sale systems (software controlling the sales, stocks, and other aspects of these businesses) for restaurants and catering. The software is customised for each restaurant/restaurant chain and the process of maintaining the diverse code base and delivering updates and debugging errors, etc. is done manually by a small dedicated team at the company. The HI Iberia product team is the provider of the POS software. The restaurant chains would not be directly involved during the project but some installation sites would be chosen as testbeds for AIDOaRt advances. HI Ibera R&D team would be developing tools based on AIDOaRt progress, integrating tools and methodologies from the project when profitable.

G. Prodevelop: Smart Port Platform monitoring

Prodevelop case study is a smart platform in charge of monitoring the activities of a port in real time, through the analysis of data coming from sensors (IoT) and information systems (legacy and external systems). This platform is a data intensive system that receives and analyses thousands of data per second using big data technologies running in virtual machines of containers on premise or on cloud infrastructures. The AIDOaRt Project will help developers in finding errors in the predefined information flows. Simultaneously, it is currently difficult to track the flow that follows the information and identify the node/process that produces errors, bottlenecks, etc. Both the detection of anomalies and the proposed visualization tools will help to faster develop more robust products.

H. Tekne: Agile process and Electric/Electronic Architecture of a vehicle for professional applications

The Tekne case study applies DevOps for developing a cloud-enabled Prognostics and Health Management System (PHM). The PHM is in charge of monitoring, diagnostics, and preventive maintenance of the on-board electronics. The system applies to the vehicle, which is an electric minivan prototype built up integrating COTS and in-house components, as well as to the payload.

I. Volvo Construction Equipment: Data modeling to support product development cost and efficiency

This use case will focus on the use of data collection and AI/ML analysis techniques to improve the model-based development process with the main goal of designing a holistic model-based enterprise approach integrating data sources into the life cycle of products. The use case involves system modeling and data modeling to support architectural analysis and system verification for a new generation of products integrating emerging technologies, namely, an autonomous hauler. The models will be based on existing functional models of traditional haulers.

J. Westermo Network Technologies: Automated continuous decision making in testing of robust and industrial-grade network equipments

WeOS (the Westermo Operating System) is the core software deployed in robust switches or routers for industry networks in harsh environments. The development process of WeOS has elements of agile kanban, continuous integration, continuous deployment and many, but not all, steps in this process are automated. By integrating smart solutions (e.g., based on AI/ML) as part of the built environment and engineering process of WeOS, it is possible to improve risk mitigation, by adopting a smart solution, and test coverage.

V. PLAN AND EXPECTED RESULTS

Fig. 3 depicts the AIDOaRt project timeline, including the main activities (architecture design, foundations, data framework, core infrastructure, and AI-augmented toolkit development) and work items (UC requirements and scenarios definition, UC development and evaluation, integration design and development).

Whenever possible, the AIDOaRt achievements will be measured using quantitative and rigorous parameters based on both metrics defined in the literature and data collected by partners during the project development. Some evaluations need to be considered with respect to the relative improvements that will be achieved when compared with already existing and applied technologies and practices in the companies. To this end, we plan to provide a reliable measurement approach so that the normal variations that might occur on design, development and operation activities are largely removed. This way, it will be easier to represent the impact of AIDOaRt results as an expression of relative improvement, which can be more easily understood.

Special attention will be paid in case that improvements provided by the AIDOaRt technologies are difficult or complex to quantify, or they have a significant subjective element affected

by the industrial domain, capabilities, or previous experience of the individuals carrying out the evaluations. Though the definition of quantifiable measures remain an important, the other criteria may be equally important since they represent key aspects e.g. ease of use, maturity, etc. that can have a substantial impact on whether AIDOaRt results will be broadly adopted by industry. As a consequence, these improvements are expected to be quantified using satisfaction measures which are typically calculated using questionnaires and interviews, where questions will be asked about certain capabilities and experiences associated with use of the AIDOaRt framework.

VI. Conclusion

AIDOaRT is a 3 years long H2020-ECSEL European project involving 32 organizations, grouped in clusters from 7 different countries. AIDOaRt aims at strengthening European excellence in Continuous System Engineering of CPSs with AI-augmentation of their engineering process The major objective of AIDOaRt is to provide a model-based framework to more efficiently support the continuous software and system engineering of CPS via AI-augmentation.

AIDOaRt intends to reach its goals through an earnest attempt to capitalize on both the state-of-the-art gained by research organizations from prior projects and the consolidated experience of the industrial partners AIDOaRt applies to and will be demonstrated on heterogeneous domains and target applications ranging from automotive, aerospace, maritime, railway etc. coming from different industrial use cases. *However, the project results will not be limited to these specific scenarios and* nevertheless, AIDOaRt aims at contributing to the creation of a cross-domain ecosystem of integrated AI-augmented solutions for System and Software Engineering support.

Acknowledgment

References

[1] H. Thompson, M. Reimann, and D. Ramos-Hernandez, *Platforms4CPS, Key Outcomes and Recommendations*. Steinbeis-Edition, 2018.

[2] D. C. Schmidt, "Guest editor's introduction: Model-driven engineering," *Computer*, vol. 39, p. 25–31, Feb. 2006.

[3] M. Brambilla, J. Cabot, and M. Wimmer, *Model-Driven Software Engineering in Practice, Second Edition*. Synthesis Lectures on Software Engineering, Morgan & Claypool Publishers, 2017.

[4] C. Ebert, G. Gallardo, J. Hernantes, and N. Serrano, "Devops," *IEEE Software*, vol. 33, no. 3, pp. 94–100, 2016.

[5] R. Jabbari, N. bin Ali, K. Petersen, and B. Tanveer, "What is devops? a systematic mapping study on definitions and practices," in *Proceedings of the Scientific Workshop Proceedings of XP2016*, XP '16 Workshops, (New York, NY, USA), Association for Computing Machinery, 2016.

[6] S. G. Charley Rich, Pankaj Prasad, "Market guide for aiops platforms, id g00378587," tech. rep., Gartner Research, 2019.

[7] Y. Dang, Q. Lin, and P. Huang, "Aiops: Real-world challenges and research innovations," in *2019 IEEE/ACM 41st International Conference on Software Engineering: Companion Proceedings (ICSE-Companion)*, pp. 4–5, 2019.

[8] *AIDOaRt Project Website*, 2021. https://www.aidoart.eu/.

[9] B. Fitzgerald and K.-J. Stol, "Continuous software engineering: A roadmap and agenda," *Journal of Systems and Software*, vol. 123, pp. 176–189, 2017.

[10] W. Afzal, H. Bruneliere, D. Di Ruscio, A. Sadovykh, S. Mazzini, E. Cariou, D. Truscan, J. Cabot, A. Gómez, J. Gorroñogoitia, *et al.*, "The MegaM@ Rt2 ECSEL project: MegaModelling at Runtime–Scalable model-based framework for continuous development and runtime validation of complex systems," *Microprocessors and Microsystems*, vol. 61, pp. 86–95, 2018.

[11] F. Bordeleau, B. Combemale, R. Eramo, M. van den Brand, and M. Wimmer, "Towards model-driven digital twin engineering: Current opportunities and future challenges," in *Systems Modelling and Management* (Ö. Babur, J. Denil, and B. Vogel-Heuser, eds.), (Cham), pp. 43–54, Springer International Publishing, 2020.

[12] M. Blumreiter, J. Greenyer, F. J. Chiyah Garcia, V. Klös, M. Schwammberger, C. Sommer, A. Vogelsang, and A. Wortmann, "Towards self-explainable cyber-physical systems," in *2019 ACM/IEEE 22nd International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C)*, pp. 543–548, 2019.

[13] *Gartner Predicts the Future of AI Technologies*, 2021 (accessed: 30.06.2021). https://www.gartner.com/smarterwithgartner/gartner-predicts-the-future-of-ai-technologies/.

[14] A. Barredo Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. Garcia, S. Gil-Lopez, D. Molina, R. Benjamins, R. Chatila, and F. Herrera, "Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai," *Information Fusion*, vol. 58, pp. 82–115, 2020.

[15] V. Dignum, "Responsible autonomy," *CoRR*, vol. abs/1706.02513, 2017.

[16] L. Leite, C. Rocha, F. Kon, D. Milojicic, and P. Meirelles, "A survey of devops concepts and challenges," *ACM Comput. Surv.*, vol. 52, Nov. 2019.

[17] B. Combemale and M. Wimmer, "Towards a model-based devops for cyber-physical systems," in *Software Engineering Aspects of Continuous Development and New Paradigms of Software Production and Deployment - 2nd International Workshop, DEVOPS 2019* (J. Bruel, M. Mazzara, and B. Meyer, eds.), vol. 12055 of *Lecture Notes in Computer Science*, pp. 84–94, Springer, 2019.

[18] C. R. Manjunath Bhat, "Augment decision making in devops using ai techniques, id g00383246," tech. rep., Gartner Research, 2019.

[19] T. B. Pedersen, "Managing big multidimensional data: A journey from acquisition to prescriptive analytics," in *Proceedings of the 6th International Conference on Data Science, Technology and Applications, DATA* (J. Bernardino, C. Quix, and J. Filipe, eds.), p. 5, SciTePress, 2017.

[20] N. Ceresani, "The periodic table of devops tools v.2 is here," June 2016. https://blog.xebialabs.com/2016/06/14/periodic-table-devops-tools-v-2/, last accessed on 30.06.2021.

[21] G. Valente, T. Fanni, C. Sau, T. Di Mascio, L. Pomante, and F. Palumbo, "A composable monitoring system for heterogeneous embedded platforms," *ACM Transactions on Embedded Computing Systems*, vol. (in press), 2021.

[22] W. Maalej, M. Nayebi, T. Johann, and G. Ruhe, "Toward data-driven requirements engineering," *IEEE Software*, vol. 33, no. 1, pp. 48–54, 2016.

[23] S. Wolny, A. Mazak, M. Wimmer, and C. Huemer, "Model-driven runtime state identification," in *40 Years EMISA 2019, May 15-17, 2019, Evangelische Akademie Tutzing, Germany* (H. C. Mayr, S. Rinderle-Ma, and S. Strecker, eds.), vol. P-304 of *LNI*, pp. 29–44, Gesellschaft für Informatik e.V., 2019.

[24] L. C. Briand, "Novel applications of machine learning in software testing," in *2008 The Eighth International Conference on Quality Software*, pp. 3–10, 2008.

[25] I. Boussaïd, P. Siarry, and M. Ahmed-Nacer, "A survey on search-based model-driven engineering," *Automated Software Engineering*, vol. 24, pp. 233–294, 2017.

[26] F. Leofante, N. Narodytska, L. Pulina, and A. Tacchella, "Automated verification of neural networks: Advances, challenges and perspectives." In press, 2018.

[27] *MegaMart2 ECSEL-JU - An scalable model-based framework for continuous development and runtime validation of complex systems*, 2021 (accessed: 30.06.2021). https://megamart2-ecsel.eu/.

[28] *Atelier B*, 2021 (accessed: 30.06.2021). https://www.clearsy.com/en/our-tools/atelier-b/.

[29] *HIPPOS*, 2021 (accessed: 30.06.2021). https://hippos.io/.