Towards AIDOaRt Objectives via Joint Model-based Architectural Effort

Bilal SAID[1], Andrey SADOVYKH [1,2], Etienne BROSSE [1] and Alessandra BAGNATO [1]

[1] *Softeam, Docaposte, Paris & Nantes, France*
[2] *Innopolis University, Innopolis, Russia*

Abstract

This paper outlines the AIDOaRt project (AI-augmented automation supporting modelling, coding, testing, monitoring and continuous development in Cyber-Physical Systems) objectives, the status, and current achievements. In particular we briefly outline one of its notable results so far, a model-based requirements engineering process that the project adopted to cope with the challenges of producing coherent joint technical results based on contributions of 32 partners. We shortly overview the process and give reference to the complete process that has been already applied and evaluated in several European Research projects. That way we intend to share the best practices in reaching ambitious objectives as targeted by the European research agenda. The project has received funding by the European Union KDT JU Key Digital Technologies Joint Undertaking, and it began in April 2021 for a period of 36 months. The Research Challenges in Information Science conference topics covered by the paper are the following: Information Systems Transformations, Model-Driven Engineering, User-Centered Design, Method Engineering

Keywords

AIDOaRt, KDT, European Research, Requirements Integration Process, Model-based Requirements Engineering

1. Introduction

AIDOaRt[2] [1] is a KDT JU Key Digital Technologies Joint Undertaking project, and it aims at supporting systems engineering and continuous delivery activities, namely requirements engineering, modelling, coding, testing, deployment, and monitoring, with AI-augmented, automated Model-Based Engineering (MBE) and Development Operations (DevOps). To achieve this goal, AIDOaRt proposes a model-based framework (cf. Figure 1) that specifies proper methods and tools to enable design and run-time data collection, ingestion, and analysis to provide tailored and efficient AI/ML solutions that are to be integrated and evaluated on concrete industrial case studies involving various Cyber-Physical Systems (CPS).

AIDOaRt aims at fulfilling three main objectives [1]: (1) to provide a model-based framework for continuous engineering of CPS, benefiting from AI-augmented automation of MDE, (2) to enhance DevOps processes by integrating AI/ML techniques in their tool chains, (3) to assist the analysis of both historical and real-time data in combination with design information in order to enhance the overall engineering process. To measure the fulfilment of these objectives, the consortium defined several ambitious Key Performance Indicators (KPI) that are evaluated at the level of each case study, as well as the overall project.

[2] https://www.aidoart.eu/

The main KPIs measure the reduction of time and effort needed to detect design problems and system deviation from design, the increase in the number of automated and enhanced DevOps phases and the increase in the number of data sources used to enable the AI/ML based optimization decisions in the systems.
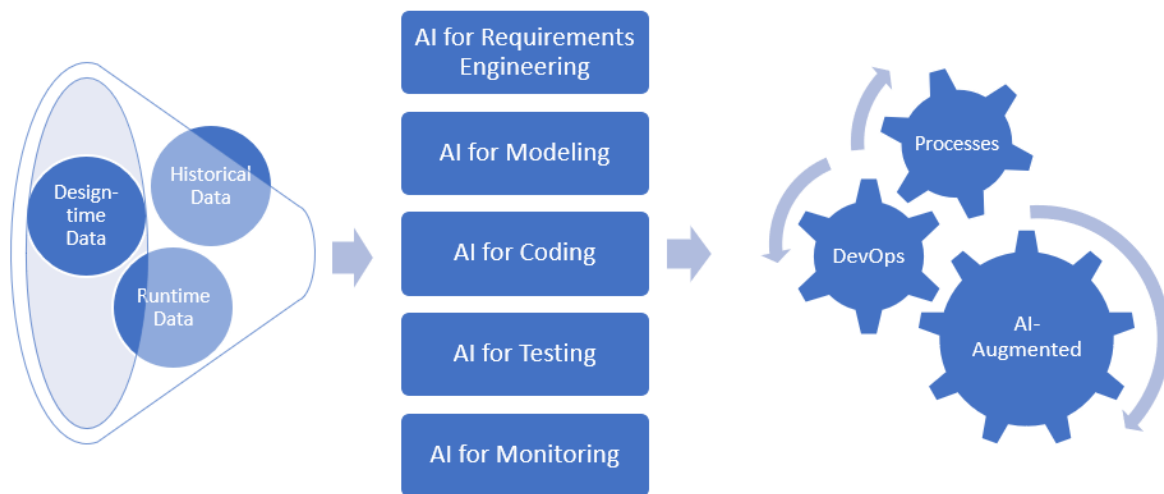

Figure 1 AIDOaRt Proposed Framework (inspired from [1])

The project involves 32 industrial and academic partners from different European countries. 10 end-user partners provide 15 industrial case studies capturing CPS of varying complexity levels and in different application domains, ranging from automotive to software and communication systems. Partners have identified in these case studies a quite varied list of 128 functional and data requirements.

The requirements largely differ in terms of abstraction levels, broadness, and coverage. We conducted a considerable analysis by separating requirements areas and partner's contributions in five key SE/DevOps activities - Requirements Engineering, Modeling, Coding, Testing and Monitoring (cf. Figure 1).

Each category was elicited by a separate working group led by experts in the related domain.

This yielded a set of 30 generic, common and high-level requirements. On a more technical level, research and industrial partners propose a set of 57 candidate solutions – building blocks to implement planned case studies. This way the project addresses a broad area of industrial requirements with regards to engineering data-centric, AI-augmented system in a model-based way. The requirements and architecture for AIDOaRt must capture 113 services to be offered by proposed solutions, with input/output data, technical and deployment constraints.

Given the large variety of case study requirements and solutions capabilities, there is an obvious need for a common framework architecture to enable collaboration between the various partners, namely among case study and solution providers.

We intended to benefit from our prior experience in previous similar large projects, namely the predecessor ECSEL project "MegaModelling at Runtime – Scalable model-based framework for continuous development and runtime validation of complex systems", or shortly MegaM@Rt2 [2]. The MegaM@Rt2 Framework proposed several interconnected tool sets for system engineering, modeling and traceability management as well as runtime analysis.

Those tool sets are interconnected to achieve the goal of linking system models with the runtime analysis of large-scale industrial systems [2]. A similar approach has been followed in AIDOaRt to propose a modular architecture with key interconnected component capabilities playing different yet complementary roles and forming together the AIDOaRt Framework. The notable contribution of the AIDOaRt approach is specific focus on the specification of data-centric and AI-augmented capabilities to be interfaced with the MBE and DevOps components.

The nature of our AIDOaRt project, in terms of number of partners and variety in requirements and proposed solutions calls for the adoption of an agile methodology to gradually define the architecture and to incrementally collect, refine and clarify the requirements, and map them to candidate solutions.

In fact, as the work in the project progresses, discussions and in-depth negotiations are taking place between case study providers and solution providers. By gradually refining the requirements in a short iteration we managed to obtain more precise and concrete requirements that in their turn drive better choices for selecting candidate solutions. This agile methodology makes it possible to integrate the newly reported updates in requirements and solutions descriptions from all partners in a flexible, simple, and traceable fashion (cf. next section). Furthermore, this model-based approach allows us to automatically generate several significant project deliverables documents with content collected and consolidated from all the 32 partners in a collaborative, distributed and seamless manner.

This paper presents the proposed methodology used to elaborate the AIDOaRt architecture and consolidate the partners requirements and solutions. It is destined to share our experience with the larger community of researchers and industries who are interested in managing the complete requirements engineering process of large research projects.

2. Application of the Model Based Requirement Engineering (MBRE) Approach in AIDOaRt

Softeam has a substantial experience of large collaborative research projects where we played a role of technical lead. In several projects (DataBio[3], REVaMP2[4], MegaM@Rt [2]), we noticed similar needs and patterns for Requirements Engineering – such as need for integration of contribution of multiple partners, need for traceability between requirements and architecture as well as similarities in the architectural concepts to be captured – tool components, interfaces, services, deployment constraints.

That information had to be captured in a coherent and homogeneous way in multiple project deliverables. Softeam is a renowned innovator in the MBE world – we consistently promote model centered engineering principles since 2000s and offer Modelio workbench. In the above-mentioned research projects, we proposed model-based tool with the goal to address the RE challenges.

In this context, we explored several modelling standards such as TOGAF/ArchiMate ([3], [4]), UML [5], and SysML [6]. In the final version, a very small subset of UML has been chosen and proved to be very efficient while providing great expressiveness and flexibility.

These results are validated in [7] and [8] where partners provided an encouraging feedback on the approach features such as scalability (e.g. large number of handled requirements), heterogeneity (e.g. partners with different types of RE background), traceability (e.g. from the requirements to the software components), and automation (e.g. requirement documentation generation) [7].

As many AIDOaRt partners had previously experience with UML, Modelio [9] and MBRE, the project technical leaders have naturally selected this approach for coordinating requirements and architectural efforts. The flexibility has been proven by an ability of the approach to accommodate new needs for concepts such as data requirements and solution architecture. Below, we outline several notable details about this approach.

Any engineering effort requires an appropriate process. In AIDOaRt, we opted for an agile approach with short interactions proposing to edit a joint model, as shown in **Figure 2**.

To accelerate the requirements gathering, we started from simple text-based tabular forms that require no prior knowledge or expertise in modelling or requirements engineering. We then verified, corrected, and auto imported the first iteration of requirements by partners into the modelling tool. In parallel, we dedicated a special effort in educating partners about MBRE and Modelio peculiarities.

After this initial phase, we iterated with short cycles of micro-tasks design and implementation. Every couple of weeks, we collected additional model content, e.g., additional descriptions on some requirements, new types of requirements or traceability mappings between the various elements of the model. That way, week after week, by small steps, all together we have iteratively enriched our model following the evolving needs of the project technical coordinators for requirements and architectural information.

---

[3] https://www.databio.eu/
[4] https://www.revamp2-project.eu/

This incremental process allows us to continuously update the data collected by our partners at any stage of the deliverable writing, until its internal review and final submission.
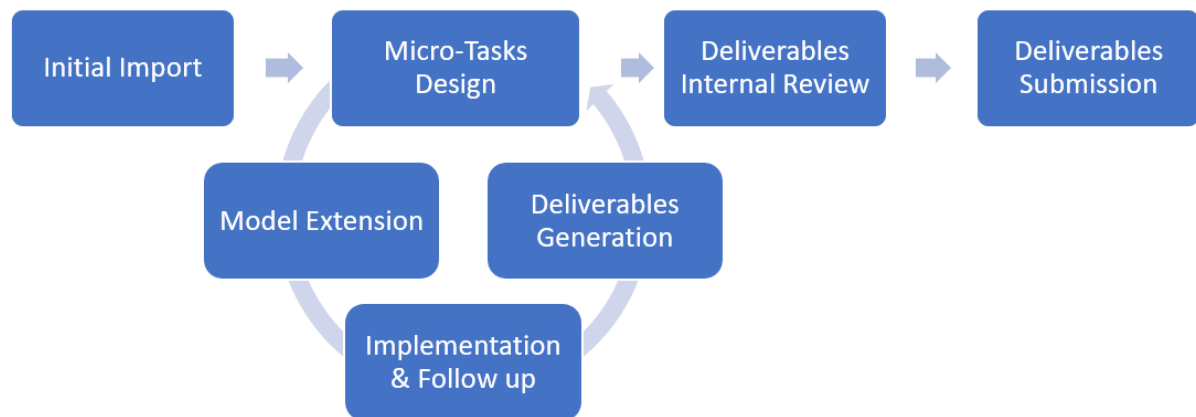

Figure 2 AIDOaRt's Agile Iterative MBRE Approach

3. Applying Modelio SaaS

The current version of the presented agile MBRE process is supported by Modelio SaaS [9], a cloud-based requirements and systems modeling tool developed by Softeam. Instead of spreading effort in editing bits and pieces of architecture numerous documents, partners work collaboratively on a single unique model representing AIDOaRt architecture including all the conceptual elements such as requirements and solution description. This model is shared through a repository where partners can connect and commit changes.

Multiple project deliverables are automatically created and updated with specifically developed document generators that export various architectural views. This shared "live" model represents a single source of truth for all the partners since this model is a unique place where the updates have to be committed. The model lifecycle covers all stages of the project from requirement inception to their validation in case study scenarios.

Meanwhile, the automated document generation guarantees a sound, traceable and up-to-date overview of AIDOaRt's architecture, case studies requirements and their fulfillment by the proposed solutions until the end of the project. It is worth noting that a significant number of project-wide deliverables have been fully created from Modelio.

To illustrate the scalability and usefulness of the approach, we would like to provide several metrics. Among the 76 users representing all project partners who are currently registered users in the Modelio repository, 60% are active users with multiple recurrent contributions. This indicates the high engagement of the partners and their adoption of our proposed approach. After ten months, i.e., almost one third of the project duration, MBRE has helped to elicit and detail 455 requirements, and a total of 6202 model elements in a raw of 1572 individual commits. This reflects the high scalability of the approach and its suitability for such large projects.

At the current stage, 3 major project deliverables[5] have been released accounting more than 200 generated pages each. The deliverables are automatically generated from the Modelio model with textual descriptions, various diagrams, and traceability matrices representing different aspects of AIDOaRt architecture. This automated generation allowed us to obtain a consistent document at every moment in time regardless of the heterogeneity and timeliness of the contributions. Furthermore, it allows to capture the most up-to-date content as provided by the partners, which continued to flow into the shared repository until the last moment before the deliverables' submission. Any other manual document aggregation approach would have resulted in considerable delays and potential difficult-to-handle inconsistencies.

---

[5] https://www.aidoart.eu/aidoart/results/deliverables

In conclusion, the MBRE approach proved to be beneficial since it allowed us faster integration of various contributions. In the beginning we experienced a brief "learning curve" period when partners had to install Modelio and get familiar with requirements modeling approach.

Through the iterations of our agile process the partners learned to appreciate the approach that helped us to drastically improve productivity and easily integrate new features into the model. The documents were generated on the go without any troubles for managing consistency of the documents and formatting. At the moment of this writing, three new deliverables are being prepared, and two are planned for the next months. All of these deliverables will be also auto generated from Modelio with content being iteratively collected using the agile MBRE approach.

4. Conclusion and Future Work

This paper presents the AIDOaRt project objectives. AIDOaRt builds a framework for AI-augmented model-based engineering. The project is in the first very important stage i.e., scoping of the project by analyzing the case study requirements and matching them with consortium capabilities. During this phase, the project produced a set of fundamental artifacts including the requirements analysis and the architecture of proposed solutions. We applied a Model-Based Requirements Engineering approach that helped us to bootstrap collaboration in the project and produce an integrated view of the project objectives in terms case study requirements and tool capabilities. This approach enabled us to interactively analyze various aspects of the project and create coherent and consistent reports with document generation.

For the future, the project intends to develop a great number of engineering solutions to address the case studies and research challenges. The approach we apply will further help us in validation and demonstration activities e.g., to specify the roadmap, KPIs and keep traceability of requirements.

Acknowledgements

References

[1] R. Eramo *et al.*, "AIDOaRt: AI-augmented Automation for DevOps, a Model-based Framework for Continuous Development in Cyber-Physical Systems," in *2021 24th Euromicro Conference on Digital System Design (DSD)*, Sep. 2021, pp. 303–310. doi: 10.1109/DSD53832.2021.00053.

[2] W. Afzal *et al.*, "The MegaM@Rt2 ECSEL project: MegaModelling at Runtime – Scalable model-based framework for continuous development and runtime validation of complex systems," *Microprocess. Microsyst.*, vol. 61, pp. 86–95, Sep. 2018, doi: 10.1016/j.micpro.2018.05.010.

[3] "TOGAF: The Open Group Entreprise Architecture Framework," *The Open Group Website*, May 04, 2018. https://www.opengroup.org/togaf (accessed Dec. 10, 2021).

[4] "The ArchiMate® Enterprise Architecture Modeling Language," *The Open Group Website*, Oct. 12, 2018. https://www.opengroup.org/archimate-forum/archimate-overview (accessed Dec. 10, 2021).

[5] "OMG Unified Modeling Language." https://www.uml.org/ (accessed Dec. 10, 2021).

[6] "OMG Systems Modeling Language (SysML)." https://www.omgsysml.org/ (accessed Dec. 10, 2021).

[7] A. Sadovykh, D. Truscan, and H. Bruneliere, "Applying Model-based Requirements Engineering in Three Large European Collaborative Projects: An Experience Report," in *2021 IEEE 29th International Requirements Engineering Conference (RE)*, Sep. 2021, pp. 367–377. doi: 10.1109/RE51729.2021.00040.

[8] A. Sadovykh *et al.*, "Model-Based System Engineering in Practice: Document Generation-MegaM@ Rt2 Project Experience," in *Proceedings of the 14th Central and Eastern European Software Engineering Conference Russia*, 2018, p. 9.

[9] P. Desfray, "Model repositories at the enterprises and systems scale: The Modelio Constellation solution," in *2015 International Conference on Information Systems Security and Privacy (ICISSP)*, Feb. 2015, p. IS-17-IS-17.