ZDMP Technical Challenge

Christian Melchiorre[a], Philip Usher[b], Tim Dellas[c], Alessia Focareta[d], and Mircea Vasile[e]

[a] *AlgoWatt, S.p.A., Via de Marini, Genoa, Italy*
[b] *Information Catalyst for Enterprise Limited, C/O Bb, Winnington Avenue, Northwich, CW5 4EE, UK*
[c] *Ascora GmbH, Birkenallee 43, 27777 Ganderkesee, Germany*
[d] *FIDIA S.p.A., Corso Lombardia, 11, 10099 San Mauro Torinese, Italy*
[e] *SIMAVI, Complex Victoria Park, Corp C4, Etaj 2, Şos. Bucureşti, Romania*

Abstract
This chapter discusses the technical approach followed in the Industry 4.0 project ZDMP. It includes the overall architecture and specifications which are the foundation for the project's development activities from both a functional and technical point of view.

Keywords 1
Service Oriented Architecture, Software Engineering Process, Microservices, RAMI 4.0

1. Introduction

The software engineering process of the project started with the identification and analysis of requirements, based on the work performed in an initial phase where business challenges were addressed. These activities were accompanied by a phase in which the users and RTD providers have worked closely together to deliver mock-ups for each envisioned application. This has been found to not only better engage users but also to facilitate more realistic and valid development priorities. Based on the requirements, and exploiting partner's knowledge, experience, and competences, a system architecture was produced, defining individual components that build up the ZDMP system, and the connections to each other and to the outside world. The system architecture has naturally led to a Functional Specification, validated through user engagement, in which functional requirements of all project components are identified, including strategic, high level functional, non-functional, and technical requirements. Coupled with the functional specification, a technical specification document was produced, defining concrete interfaces between ZDMP components, protocols, and class/package structures. In parallel to all these activities, cross-cutting standardization activities define the contribution of ZDMP and the usage of components with regards to the latest standardization efforts.

2. Requirements and User Mockups

The requirements analysis is the borderline between the business/requirements aspects and the software engineering process of the project. The requirement specification delivered as result of these activities consider different aspects. Among these, functional requirements from the target user and provider scenarios and those arising from the preliminary scenarios defined within the pilots. Moreover, it considers the state-of-the-art analysis and partners' further know-how and expertise. This primarily included the analysis done in the initial phase of the project addressing the business challenges, where industry scenarios and Use Cases where analysed, mapped, and aligned to existing Zero-defects Manufacturing Reference Models, resulting in the delivery of a list of user/system needs and features for the industrial pilot implementations.

Proceedings of the Workshops of I-ESA 2020, 17-11-2020, Tarbes, France
EMAIL: christian.melchiorre@softeco.it (C. Melchiorre); philip.usher@informationcatalyst.com (P. Usher); dellas@ascora.de (T. Dellas); a.focareta@fidia.it (A. Focareta); mircea.vasile@siveco.ro (M. Vasile)

Requirements gathering was one of the main activities conducted for the definition of a project's scope and the basis of project design. Several iterations have been performed in order to describe functionalities of the system under development guiding the implementation to fulfil and accomplish the scope of the project. The followed methodology includes all actions and best-practices as defined by the Capability Maturity Model Integration (CMMI) process [1]. Among these, elicitation of needs, transformation into user requirements, refinement of user requirements into software requirements, analysis of requirements, stakeholder needs and constraints balance, and validation of requirements by all major stakeholders.

The issued Requirement Specification document includes the identified requirements for all components, divided into strategic as well as high level functional and technical requirements. Additionally, as stated by the Project Management Institute's Project Management Body of Knowledge ([2]), the document includes a traceability matrix that links the product requirements to their source (and, additionally, to the component implementation that satisfies them). The layout and the content allows the traceability of ZDMP project requirements. This includes the connection between the User Requirements and its source (e.g. interview, document), as well as the connection between the User, System, and Software Requirements. All user requirements are covered by software requirements. It is expected that this traceability will be extended during the future development activities to trace and interconnect the requirements with product design and development evidences, and with test scenarios and test cases.

Coupled with this work, the users and RTD providers worked closely together to produced User Mock-ups for each application proposed during the industrial scenarios and Use-Cases analysis. This has been found to not only better engage users but also to facilitate more realistic and valid development priorities. One vital aspect, especially in the open and inter-cooperating environment of ZDMP, is "standards engagement" to ensure interoperability and a level playing field.

User Mock-ups consist in a graphical visualization illustrating the initial expectations of the user interface for each application from the users' perspective. The proposed applications are thus enriched with a higher level of detail, defining in each screen the input and output information, the users' identities and permits, as well as the physical supports where the application will be used. Even if this work was performed very early in the project, the graphical interface requirements definition, the functionalities description, the coherence of the mock-ups with the actors, and roles verification have only been possible after a detailed drawing of the expected graphical interfaces. The availability of these is also extremely useful for the technical partner not only as an 'exact' reference but also to help understand how users will approach and use ZDMP in practice.

3. Global Architecture Definition

The first step to transform the requirements and the User Mock-ups into a global architecture consisted in the identification of the main components and their interactions. ZDMP platform's architecture is based around a Service Oriented Architecture (SOA) and microservices approach. ZDMP components are accessed as services mainly through RESTful APIs or through a Service and Message Bus.

The ZDMP architecture has four tiers which are composed of components: a Developer Tier (design-time), an Enterprise Tier (use-time), a Platform Tier (run-time), and an Edge Tier (distributed run-time). All tiers except the Developer Tier are based on the RAMI 4.0 architectural model [3]. This is illustrated in the following Figure.

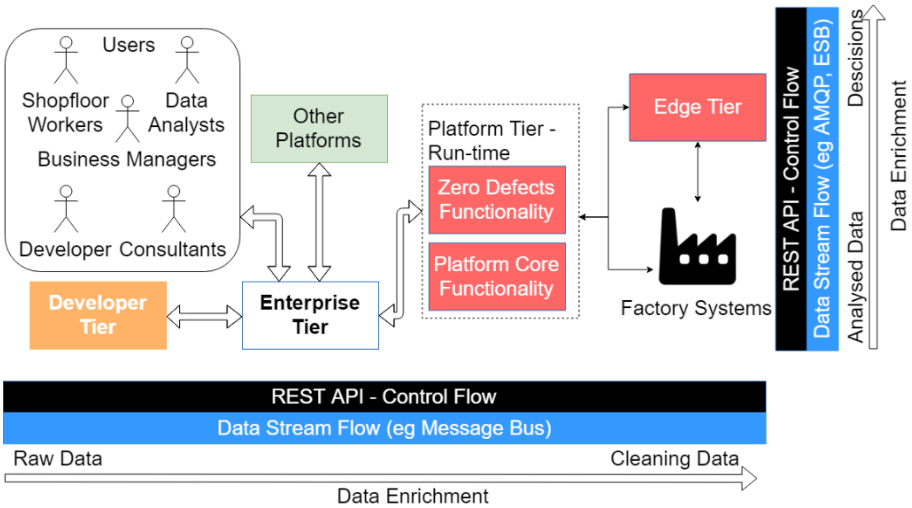

Figure 1: ZDMP Architecture

The **Development Tier** focuses on components that are needed to create applications used to solve concrete use-cases. These Development Tier components are installed separately to the platform on development machines. The aim is to create containerized applications from these components that can be uploaded to the ZDMP Marketplace and from there they can be obtained by users. The **Enterprise Tier** includes higher level components that deal with aspects of running the containers and interactions between users of several types. It represents use-time components that are needed to organize, run, and set up the platform but not provide any of the core ZDMP functionalities.

The **Platform Tier** is split into two major sections. This includes the ZDMP functionality in the form of applications and supporting components. It also includes core components that facilitate the use of data and organizing processes within the system.

The **Edge Tier** is needed to facilitate distributed computing and to improve performance of the platform by bringing certain aspects of data processing close to the sources of data. It also acts as place where data enters the platform..

4. Functional and Technical Specification

After the identification of the global architectural principles and the definitions of the main components that build up the ZDMP platform's architecture, the next step consisted in the definition of the functions of all necessary software components needed to implement the Use Cases and fulfill the identified requirements. In the resulting Functional Specification document, an in-dept definition of these functions and their dimensions What, Where, When, by Whom and Why is provided. This is done in close resemblance to user stories, to more closely inspect the function stakeholders, the context the function operates in, as well as timely and specially restrictions in terms of process (e.g. this function needs to be activated at the customer site close to the machine after a particular other function was executed by a worker with manager access rights).

ZDMP Functional Specification is divided into the functionality and interactions that the user has with each ZDMP component. The functional analysis per component and zApp has two perspectives each:

- Behaviour and Functionality: Containing a story map with the features and functionality offered and the user stories that need to be developed to implement that functionality
- Interaction descriptions: Describing for each component the set of interactions with other ZDMP components and users and describing the exchange of information flows critical for a unified ZDMP platform.

The main description presents the feature of software from the point of view of the subject who expects the feature. The subject is not restricted to a 'human' ZDMP user (eg an operator or developer) and can be any entity with a behaviour, e.g. the component being described, another component, etc.

Features include acceptance criteria – a checklist that determines when the feature is considered completed. The acceptance criteria are expressed from the point of view of the user listed in the Who-part and provides a detailed description of the criteria by which user stories should be evaluated and validated. Features have a unique ID (e.g. T55F001). Deadlines list the project month when the feature should be completed (e.g. M18, M24 and M30).

The ZDMP Technical Specification defines concrete interfaces between ZDMP components, protocols, and class/package structures. This includes definitions of methods, parameters, return values, and error handling for each component and interface. It contains data models and concrete data schemas to be used on the source code level and help select the (software) technologies to be reused and applied within the project, based on a study of possible technologies. Moreover, it is used to define the missing functionalities and implementation needs, which are the foundation for the work to be performed in the further RTD work packages. The document is delivered as an online dynamic reference, to facilitate the further software engineering steps. The online approach will also facilitate sub-call partner inclusion. Since this represents partners IPR for exploitation, this specification was originally defined as confidential. However, the partners are discussing (and still to conclude) the expectation to make this, or part of this, public since it will facilitate, or may be necessary, for the sub-calls.

5. Standardization Issues

One vital aspect, especially in the open and inter-cooperating environment of ZDMP, is “standards engagement” to ensure interoperability. Strategic motives for companies to participate in standardization activities include the ability to design industry friendly regulations, enforce own content, prevent formal standards from conflicting with own interests, solve industry specific technical problems, and acquire competitive advantages through a head start in knowledge.

The goals of ZDMP standardization activities are to connect to standardization forums and to monitor the project to ensure a compliance of the project results in existing standards. Standards support three of the main objectives in ZDMP: Extensibility, interoperability. and openness. Companies around the world use standardized interfaces, enabling the communication and thus the collaboration between different software packages or whole machines. The definition of standardized interfaces is open in a way that it is accessible by everyone which enables different stakeholders to create compatible programs or devices. By supporting the most used interfaces, ZDMP will be able to run in many factories, thus accessing a huge market.

ZDMP creates an open ecosystem, where developers can offer additional modules to extend the range of functions. A standardized interface of ZDMP with these modules could ensure an open and transparent environment, giving developers security and minimizing errors. This can be especially valuable and reassuring in the ZDMP sub-calls as well.

To utilize standardization most efficiently, ZDMP sets the following standardization goals:

- Interaction: Establish contact to relevant groups to receive and provide information
- Compliance: Monitor the use of standards in the project to foster compliant results
- Input: Engage in standardization to exploit project results

Upon an analysis of the current standardization landscape with respect to ZDMP, this deliverable describes how these goals can be achieved and proposes certain actions in the recommendation section. Since different project partners are already members in different standardization consortia, a solid interaction with standardization is already established and will be improved. The project partners provided a first evaluation of which standards are important and which standards will be used. Topics, suitable for input to standardization were identified and ZDMP expects to achieve the creation of one CEN Workshop Agreement (CWA) and is motivated to create up to three CWAs. This includes a CWA to be envisaged in cooperation with the H2020-projects eFactory and Qu4lity.

6. Conclusions

Several activities are aimed to tackle the technical challenge of a project like ZDMP. These include the elicitation of requirements from all the relevant stakeholders in order to establish what are the main needs pertinent to the various product lifecycle phases, and the use of the results of these activities to define a global architecture for the ZDMP platform, including the definition of components and their functional and non-functional capabilities. This paper describes the methodology, main approach and the results of such activities performed in the ZDMP project.

7. Acknowledgements

The research leading to these results received funding from the European Union H2020 Program under grant agreement No. 825631 “Zero Defect Manufacturing Platform (ZDMP)”.

8. References

[1] The Capability Maturity Model Integration (CMMI) Institute URL: https://cmmiinstitute.com/.
[2] The Project Management Institute. (2013). “A guide to the Project Management Body of Knowledge (PMBOK® guide)(Fifth Edition)”. Project Management Institute.
[3] Deutsches Institut für Normung e. V. (2019) Reference Architecture Model Industrie 4.0 (RAMI 4.0) English translation of DIN SPEC 91345:2016-04.
[4] Coutinho, C., Lopes, L., Viana, V., Pape, D., Klasen, G., Von Halem, B., Garcia Perales, O., Stellingwerff, L. Stam, A., “An open environment for development of manufacturing applications on vf-OS”, Enterprise Interoperability: Smart Services and Business Impact of Enterprise Interoperability, 2018, p. 107-114.
[5] Manteigas Da Cunha, L., Stellingwerff, L., Stam, A., “A Novel Approach to Software Development in the Microservice Environment of vf-OS”, Enterprise Interoperability: Smart Services and Business Impact of Enterprise Interoperability, 2018, p. 115-119