Reflecting on APPsist, a Service-based Architecture for AI-based Support on the Shop Floor

Carsten Ullrich[1] and Christoph Igel[1]

DFKI GmbH, Germany,
carsten.ullrich|christoph.igel@dfki.de

Abstract. In this contribution, we reflect on the APPsist project, in which we designed and developed an architecture for context-sensitive and intelligent-adaptive assistance systems for knowledge and action support on the shop floor. We describe and comment on the socio-technical perspective taken in APPsist, which integrated technical, organizational and human dimensions, and on the technical approach, which aimed at developing a general architecture easily applicable in different scenarios and making use of existing technical infrastructure.

Keywords: workplace-integrated learning, assistance, support, AIED

1 Overview on the APPsist Project

The goal of the APPsist project was to develop a new generation of mobile, context-sensitive and intelligent-adaptive assistance systems for knowledge and action support on the shop floor. It was funded by the German Federal Ministry for Economic Affairs and Energy (BMWi) for a period of three years (2014–1016) under the "Autonomik" initiative for INDUSTRIE 4.0. The project focused on the skills and competencies of the human operators on the shop floor and attempts to compensate for any skills that may be lacking with respect to performing tasks at the workplace (action support). In addition, knowledge-support services facilitate the continuous expansion of staff expertise through the acquisition of knowledge and skills in relation to production, product, and process. The aim was to promote the professional development of the staff so that they can gradually start to perform more demanding tasks and serve as a counterbalance to the demographic change and the shortage of skilled workers. This support includes the setup and operation of a manufacturing unit in the production process, as well as the preventive maintenance, maintenance, and troubleshooting.

2 Motivation

Industry 4.0 and the set of technologies it subsumes gives rise to challenges regarding the training and learning of employees but also offers new possibilities for technology-enhanced learning. The goal of the APPsist project was to investigate these challenges and chances from a comprehensive perspective, integrating technological as well as organizational and human-oriented viewpoints.

Todays workplace on the shop floor (the area of a factory where operatives assemble products) is highly demanding [1]. The foremost goal is to maintain productivity to fulfill customer orders by producing the required number of products. The environment is a highly complex one: the machines become ever more complex, as do the products. Furthermore technological innovation results in new materials and new technologies being used in production and for processing and assembling products. Last but not least, a decreasing workforce requires employees to become more flexible and master larger number of skills, for instance to be able to stand in when colleagues are not available. This requires to use machines that are not the primary area of expertise. As a consequence, the employee is under constant pressure to solve problems occurring on the shop floor as fast as possible, and simultaneously to improve his work-related knowledge, skills, and capabilities. The question thus arises, how knowledge and assistance services can support the individual human operator while working and learning.

On the other hand, this digitized environment is filled with sensors controlling the production process and thereby offering a digital representation of the actual physical state of the shop floor. Here, the question arises how knowledge and assistance services can use this sensor data for improving their support.

3 Approach

From the very beginning, research and development in APPsist followed a socio-technical approach, which integrated technical, organizational and human dimensions [2]. The organizational perspective was represented by a scientific institute that focused on the organizational implementation (integration) of the developed technology into a company. The human/personnel perspective was represented by having as an associated project partner Germany's largest trade union as well as work councils of one of the companies of the project. Regular (online and on-site) meetings attended by partners from all three perspectives discussed the technological development. In addition, workshops with the target groups served to present and discuss project results (mock-ups and prototypes).

On reflection, the project benefited significantly from the socio-technical approach. Even though sometimes the joint work was difficult due to problems in understanding each others vocabulary and to expectations difficult to meet, the partners became able to reach a basic understanding of the respective perspectives. The union partner remarked that a more methodological and systematic approach might have been more beneficial, but it is unclear what such an approach could have looked like.

The technological approach was based on a micro-service architecture [3]. Micro-services are fine-grained, replaceable and self-contained and the features are provided by lightweight application interfaces. These features are surely desirable, yet, in practice in APPsist they turned out to be difficult to achieve.

While early-on in the project, we compiled a list of functionally distinct services, these were not self-contained but highly dependent on each other. In our opinion, this is mainly due to lack of time, and a revised architecture might result in true micro-services.

We based the central data-structures on standard technology. We used xAPI[1] for storing the users' interactions with the environment. The process model, i.e., the activities the operators have to perform in order to achieve a task, is specified in BPMN (Business Process Model and Notation [4]). The shopfloor domain, i.e., the domain concepts and entities and their relationships are modeled in the description language OWL [5] and stored in a semantic database (a triplestore). This model defines an unambiguous vocabulary used for communication between the services and serves as the basis for the reasoning processes of the adaptive services [6]. For instance, the measure service determines the measures applicable in the current situation through a semantic database query (SPARQL [7]). SPARQL offers a relatively easy to master yet expressive query language. A combination of Java code and SPARQL queries were sufficient to encode the knowledge for selecting relevant work procedures and content [8].

In hindsight, BPMN was a good choice for modeling the work procedures, as it is easy to understand and known by some of the groups involved in the actual modeling. However, as an internal data structure for controlling the interaction with the system it is slightly to restricted. There, other approaches such as behavior trees [9] might offer more flexibility.

4 Core contributions

The key contribution of the research was a service-oriented architecture for support of human operators on the shop floor that was applicable in several distinct settings. The use cases covered a small-, a medium- and a large-sized company, with the following pilot scenarios:

The *small-sized company* produces complex customer-specific tools and devices for car manufacturers and their suppliers. The pilot scenario focuses on installation and use of devices (milling machines). The target audience was high-skilled experts.

The *medium-sized company* produces customer-specific welding and assembly lines for car manufacturers. The pilot scenario focuses on error diagnosis and correction in the customer-specific machines. The target audience were customers of the company.

The *large-sized company* produces pneumatic and electric controllers for the automation of assembly-lines, which are used in customer-specific products as well as in their own production. The pilot scenario focuses on maintenance and repair, in particular outages (replacement of adhesives). The target audience consisted of un- and low-skilled workers.

The APPsist system was applicable in all three scenarios, without any modification and only minor specializations. Each scenario required its own description

[1] https://github.com/adlnet/xAPI-Spec

of the specific shop-floor configuration, i.e. instances of the classes represented in the APPsist ontology. The pedagogical knowledge that controlled the selection of content adapted to the individual user and context was the same for all three scenarios [8].

Furthermore, APPsist showed that the idea of Smart Services, i.e. using existing infrastructure and data to implement new functionalities does hold for Smart Production. No additional sensors were required, instead APPsist made use of existing hardware in order to perform its adaptive support. Similarly, to a large extend, the APPsist services reused content that was already available within the companies.

However, the reuse of existing machinery and content required a high amount of manual work, which was not scalable. We had to write adapters for machines in order to use sensor data such as the internal state of a robot cell (is it running, shutdown, or recently reseted, as well as the state of physical items, such whether a door of a station is open or closed, to give a few examples). We also had to describe existing content with metadata and first and foremost author the BPMN descriptions of the work procedures.

5 Practical impact

From a research perspective, the key impact is the proof of concept that generally applicable architectures can support human operators in domains such as industrial production. First, it is possible to model the shop floor domain and the pedagogical knowledge in such a way that it can be applied to very different use cases. Second, assistance services can use sensor data from existing infrastructures to provide individually adapted support.

From the perspective of the involved companies, the key impact is a system that supports their human operators in the pilot scenarios. In the small and the large company, the APPsist is still in use.

6 Open issues

Several issues were left out. First, the learner model, that is the support-relevant information about the user, is rather shallow. APPsist's learner model stores how often the user interacted with work procedures, machinery and content items, as well as permission (the work procedures an operator is allowed to perform, specified by his/her supervisor). There is only limited usage of competencies and skills. It came to a surprise to us that the system provides helpful support even without modeling of competencies, even more so as in the beginning of the project one partner developed a rather comprehensive competence model. Yet, during the project, we realized that it was very difficult to match the competency model to the information actually available in the companies. For the small company, such information was in the head of the supervisors, for the large company, some of the information was “externally” available, but in various formats and degrees of detail. Also, it turned out that for the scope of the pilot scenarios, permissions were the decisive factor in selecting adequate work procedures.

Secondly, evaluation did not go beyond usability. We measured the usability of the services that use the rules using the System Usability Scale (SUS, an established industry standard) [10] and AttrakDiff [11]. Six employees of each industry partner received a number of tasks to solve using the system and were asked to think aloud while working on the tasks. Afterwards, they scored the system according to the SUS criteria, yielding an average score of 86.9, which is a very high score (a rating of excellent) and comparable positive score on AttrakDiff. Also the analysis of the think-aloud protocols did not show any problematic points. The results for all three partners were comparable. However, more long-term evaluations with larger numbers of participants have to follow to better understand the effects of the system after longer periods of usage.

7 Follow on

In follow on projects, we addressed the problem of authoring and also applied the APPsist architecture to a different domain. The project DigiLernPro focuses on semi-automatically-generated digital learning scenarios for supporting employees in industrial production [12]. An adaptive authoring software enables operators and trainers to author work procedures very quickly. A wizard-like approach ensures that the authors, even if not qualified trainers, input pedagogically relevant information.

In the ALINA project [13], we applied the APPsist architecture to the domain of interdisciplinary emergency admissions. While some services remain applicable without changes, other require substantial modification. Foremost, the medical partners rejected the strict linear sequence through the work procedures and demanded a free navigation that allows the user access to each step from each step. Such a requirement contradicts the semantics of BPMN (or requires explicitly representing the transitions, which make the BPMN difficult to author and read). Therefore, we plan to reimplement the assistance service.

More generally speaking, APPsist is an example of an immersive learning system in the sense of using data sources available in a physical environment. Thus, it faces the question of integrating data from a variety of sensors and systems. How to do this, is a general research question raised e.g. by [14] under the term of "community learning analytics."

8 Future work

Highly relevant is the problem of scalability. Architectures supporting problem solving and knowledge acquisition will only find widespread application if the cost of applying them to a new setting is reasonably low. Currently, integrating new content, processes and production machines into APPsist, requires manual input of the metadata, machine instances, etc., resulting in significant costs. Here, methods of information extraction that analyze existing documents might allow automating the ontology creation, instance creation and metadata annotation, and thus enabling low-cost, scalable support of human operators. Also, the question of how to integrate the sensors, actors, data and services, i.e. technology of the Internet of Things, is highly relevant for the future.

References

1. Mavrikios, D., Papakostas, N., Mourtzis, D., Chryssolouris, G.: On industrial learning and training for the factories of the future: a conceptual, cognitive and technology framework. Journal of Intelligent Manufacturing **24** (06 2013)
2. Hirsch-Kreinsen, H.: Wandel von Produktionsarbeit – "Industrie 4.0". Soziologisches Arbeitspapier 38, Technische Universit?t Dortmund, Wirtschafts- und Sozialwissenschaftliche Fakult?t, Dortmund (January 2014)
3. Newman, S.: Building Microservices. O'Reilly Media, Incorporated (2015)
4. (OMG), O.M.G.: Business Process Model and Notation (BPMN) Version 2.0. Technical report, Object Management Group (OMG) (jan 2011)
5. van Harmelen, F., McGuinness, D.L.: OWL web ontology language overview. W3C recommendation, W3C (February 2004)
6. Ullrich, C.: An ontology for learning services on the shop floor. In Sampson, D.G., Spector, J.M., Ifenthaler, D., Isaías, P., eds.: Proceedings of 13th International Conference on Cognition and Exploratory Learning in Digital Age, IADIS Press (2016) 17–24
7. Seaborne, A., Prud'hommeaux, E.: SPARQL query language for RDF. W3C recommendation, W3C (January 2008) http://www.w3.org/TR/2008/REC-rdf-sparql-query-20080115/.
8. Ullrich, C.: Rules for adaptive learning and assistance on the shop floor. In Sampson, D.G., Spector, J.M., Ifenthaler, D., Isaías, P., eds.: Proceedings of 13th International Conference on Cognition and Exploratory Learning in Digital Age, IADIS Press (2016) 261–268
9. Millington, I., Funge, J.: Behavior Trees. In: Artificial Intelligence for Games. 2nd edn. Morgan Kaufmann Publishers Inc. (2009) 334–370
10. Brooke, J.: SUS: A quick and dirty usability scale. In Jordan, P.W., Weerdmeester, B., Thomas, A., Mclelland, I.L., eds.: Usability evaluation in industry. Taylor and Francis, London (1996)
11. Hassenzahl, M.: Hedonic, emotional, and experiential perspectives on product quality. In: Encyclopedia of human computer interaction. IGI Global (2006) 266–272
12. Schütze, G., Averbeck, I., Finken, J., Freith, S., Ullrich, C.: Implementierung eines Aufnahmewerkzeugs für die semi-automatische Erstellung von Lernszenarien. In Zender, R., ed.: Proceedings of DeLFI Workshops 2016 co-located with 14th e-Learning Conference of the German Computer Society (DeLFI 2016). CEUR Workshop Proceedings, CEUR-WS.org (2016) 195–201
13. Dietrich, M., Ullrich, C.: Personalisierte lernumgebungen in der interdisziplin?ren notaufnahme. In: Proceedings of Fachkonferenz MEDIZIN.BILDUNG.DIGITAL. (2017)
14. Koren, I., Klamma, R.: Community learning analytics with industry 4.0 and wearable sensor data. In Beck, D., Allison, C., Morgado, L., Pirker, J., Khosmood, F., Richter, J., Gütl, C., eds.: Immersive Learning Research Network: Third International Conference, iLRN 2017, Coimbra, Portugal, June 26–29, 2017. Proceedings, Springer International Publishing (2017) 142–151