Mobile Staff Planning Support for Team Leaders in an Industrial Production Scenario

Sönke Knoch, Melanie Reiplinger
German Research Center for Artificial Intelligence
Research Department Intelligent User Interfaces
Saarbrücken, Germany
Email: [first name].[last name]@dfki.de

Rouven Vierfuß
Miele
imperial-Werke oHG
Bünde, Germany
Email: rouven.vierfuss@imperial.de

***Abstract*—Team leaders in industrial production scenarios face the problem of short-termed reactions to unforeseeable events in a factory. Adjustments in staff planning require many calculation and coordination efforts that consume valuable time. A mobile application is presented that assists team leaders in this challenging task. The selection and identification of relevant information, its case-specific processing and visualization as well as the persistence of worker data in a semantic network form the cornerstones of this work in progress.**

Keywords–Decision support; Staff planning; Mobility; Domain model.

I. Introduction

The developments towards Industry 4.0 [1] let the gap between real and virtual worlds melt and foster the utilization of the potentials of the Internet of Things [2]. The future industrial production aims at highly individualized products and an increasing flexibility of production processes. These developments involve a challenge for staff planning engineers who have to answer this flexibility in production with an adaptive staff planning strategy. Rising quantity of data and complexity makes this planning process even harder.

The coordination between planning engineers—in this scenario called team leaders—to find additional qualified workers within the factory consumes time and is a difficult task. To assist teamleaders in this process, the concept of a planning support system is suggested. In a defined scenario, a mobile application will support team leaders by visualizing the current worker-to-production-line allocation and interconnecting the team leaders to coordinate personnel placement in an efficient way. Therefore, information from several distributed information sources is prepared and presented on a mobile device. Relevant information dependent on the user's role and specific context of use is shown. Human ressource allocation can be edited and optimized directly using the user interface.

In Section II, an overview of related work is given and the distinction to the suggested approach is drawn. Section III describes the industrial scenario that is taken as a basis for the considerations made in the following sections. The IT-infrastructure necessary to run the system is presented in Section IV. In Section V, the domain model that gathers and stores worker profiles and supports the system in its task is developed. Section VI shows mockups of the mobile application. Finally, Section VII contains a discussion of the suggested approach and gives an outlook on future work.

II. Related Work

Current staff planning software is implemented as stand alone solution or integrated in enterprise resource planning (ERP) or manufacturing execution systems (MES). Time tracking is commonly part of the MES whereas time management can be part of both, MES and ERP [3, 199-212]. Most of these staff planning or workforce management systems focus on the scheduling and optimal resource allocation task. User interfaces present timetables with a view over a planned period in a very functional way. Some of these software providers offer mobile applications that allow access to the staff planning systems. These systems lack adaptive context- and role-specific information processing. The collaboration and coordination aspects tackled in this work are neglected. Additionally, the installation, application and maintenance costs, especially related to MES and ERP systems, are too high for many medium-sized and small factories.

In research, the project ENgAge4Pro [4] focuses on age-appropriate staff planning. Physical attributes, such as body weight and height, are regarded therefore. The research project EPIK [5] focuses on the optimal allocation of resources to enhance efficiency. A mobile application was developed that supports the worker with context-specific information. The research project KapaflexCy [6] covers short-term production scheduling. A mobile application allows to send employment requests to workers. After receiving these requests, the workers coordinate the takeover of the employment themselves. From a hierarchical perspective, this is a bottom-up approach.

Unlike the project ENgAge4Pro, the main concern of the staff planning support system is not ergonomics. In contrast to the EPIK project, the system suggested is developed to support team leaders in their planning task. Therefore, more attention is paid to the appropriate presentation of relevant information on the device. Optimized resource allocation will be included in form of an additional function (not the object of research). Compared to the KapaflexCy project, the team leader application developed here implements the allocation of employment in a top-down way. Nevertheless, worker-related information can be used to provide feedback for their work.

III. Scenario

The production of kitchen appliances in Germany is facing several challenges. Companies require the ability to produce their products under optimum cost, flexibility due to rising variants and a competitive market. Therefore, it is necessary

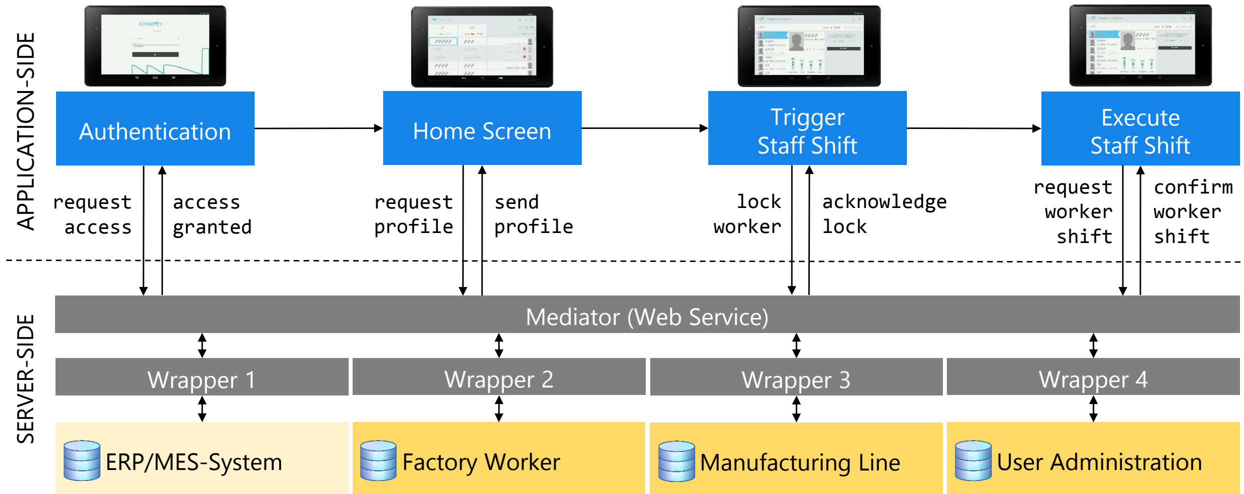

Figure 1. Staff switching process and involved distributed data sources.

that the manufacturing industry makes efficient use of resources and energy in order to keep the high-cost country Germany a competitive production location. Manufacturing of steam ovens in Imperial/Miele plant floors follows the “Miele Value Creation System”. Multiple U-shaped production lines for diverse product classes allow for highly flexible handling of varying production programs. Each steam oven is assembled by a single worker in a one-piece-flow setting, which entails high responsibility and a complex work content for all employees.

Once per week, the plant’s foremen and team leaders plan the production on the shop floor level, assigning resources and capacities to production orders. Detailed planning is done on a daily basis, considering the production program and availability of workers. In case of unexpected staff shortage or modified production volumes at short notice, the team leaders re-assign available workers to production orders and assembly stations, also across assembly lines. The process of staff planning is demand-oriented and flexible, and quickly becomes complex and time consuming while trying to meet the demands of multi-variant production scenarios with varying production programs, small lot sizes on multiple lines and customer-individual products. Furthermore, team leaders want to foster a broad skill set in all employees by organizing a rotating assignment of workers to varying tasks while, at the same time, the high quality standards of Miele need to be guaranteed by intense training on each particular product class.

A great deal of experience is needed in order to make the right decisions, and an adaptive assistant system that transparently combines data from production orders, human resource management and the plant floor in order to support decision-finding could significantly facilitate and speed up the daily staff planning process.

IV. IT-Infrastructure

To tackle the challenges described in Section III, we developed an application that supports the team leader in adjusting the daily routing when specific events occur. This kind of ad hoc planning involves heterogeneous information sources that need to be prepared and consulted in order to support reasonable decision making. The integration of these sources is discussed in Section IV-B. The integrated information needs to be preprocessed for the specific context and user role. The development of a suitable user interface for the industrial planning scenario is discussed in Section IV-A.

A. Application-side

Figure 1 shows the process that appears when a worker is scheduled. The *Authentication* forms the entry point to the application. The application loads role-specific profiles for each user. As each team leader manages different manufacturing lines, the respective lines are loaded and currently allocated workers are presented on the *Home Screen*. If the system registers any deviations from the planned schedule, it visualizes warnings and encourages the team leader to act in an ad hoc manner. One common reaction to hold the planned quantities at the end of the day is the search for suitable workers. For that reason, the application suggests suitable workers for a respective production line and product. If the team leader selects one of these workers the *staff shift is triggered*. To avoid multiple allocation of the same human resource, the selected worker is locked for 2 minutes. During this time interval, the team leader is able to execute the staff shift or search for another worker. If it was decided to schedule the selected worker he or she *executes the staff shift*. The team leader who supervises the respective worker is informed and can deny or confirm the worker shift. In the latter case, the shift is persisted and written to the database.

To compute the qualification of a worker (w) for a respective manufacturing line (l) and product (p), three parameters were defined:

$experience(w, l, p)$
Total time by a worker.

$quality(w, l, p)$
Defective pieces per shift by a worker.

$productivity(w, l, p)$
Pieces per shift by a worker.

These parameters are visualized in the detail view of the front-end described in Section VI and allow the team leader a vague estimation of the worker's skills. The ranking function $rank(w, l, p)$ forms a weighted aggregation of these parameters and represents the skill level of each worker on a scale between 0 and 5. These weights are dynamic and adapted to the specific use case.

B. Server-side

The information that is necessary to support the planning process of a team leader as described in Section IV-A is located at four different points that are sketched at the bottom of Figure 1. In the present case, the information system can be described as a multi-computer, partitioned, distributed, shared nothing system. Thus, a suitable strategy to integrate the information has to be selected. To preserve the autonomy of the sources, a virtual integration strategy was selected which leaves the data at the sources. This kind of on-demand integration in a decentralized manner enables us to keep the system design easy to extend and to transfer data only when needed from solely relevant sources. A mediator-based approach was chosen to realize the virtual integration system. The *Mediator* provides an interface implemented as a Web service and communicates with the application. It lies in the responsibility of the mediator to provide a structural and semantic data integration. *Wrappers* are implemented for each information source to overcome the heterogeneity on the data level and to enable the data flow between mediator and sources.

Focusing on the four data sources at the bottom of Figure 1, the *ERP/MES-System* forms the only system that is already existing and available in a common factory. An ERP or MES system in the considered scenario provides—in collaboration or separately—access to time tracking and management data. The domain model for the *Factory Worker* encodes the worker's skill matrix and working history. It is described in detail in Section V. The domain model of the *Manufacturing Line* describes the process steps and involved manufacturing equipment and allows estimations about quantities that can be achieved when a specific worker is scheduled on that line. In combination with processing times it can be used to calculate optimal resource allocations with algorithms from the operations research field. The factory worker model constitutes new input data that allows new forms of optimization based on the user model. The model of the manufacturing line is not part of this work and is developed separately. The *User Administration* is responsible for authentication and lock requests.

V. Domain Model of a Factory Worker

The information about workers' skills and experience that is needed by the mobile application in order to generate useful recommendations is taken from a worker model. The worker model represents semantic relationships between the horizontal and vertical roles of an employee in the factory (i.e., his tasks, work content, and his position within the staff hierarchy) on the one hand, and his skills (e.g. work experience) and individual requirements on the other hand. Examples for the latter are an employee's handedness, language skills, allergies against specific materials, or an inability to lift heavy weights or to distinguish colors. A small excerpt of the worker model

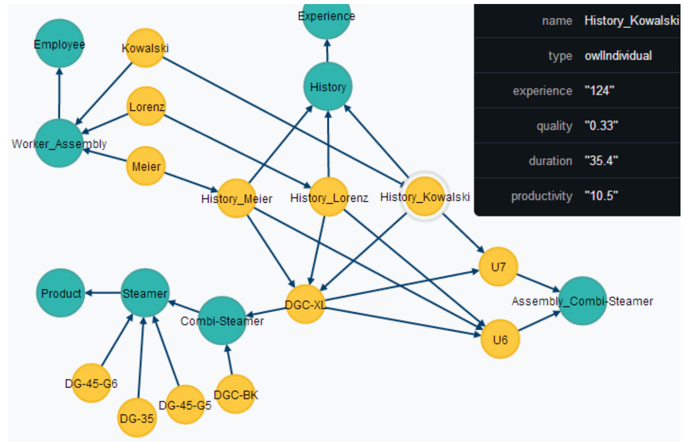

Figure 2. Snippet of the worker model.

is visualized in Figure 2. Here, the qualification of workers is encoded within the *History*-nodes of a semantic network between products, assembly lines, and employees. The application can, for instance, query the model for workers that are experienced assemblers of product p at assembly line l, and rank them using weighted aggregation as described in IV-A. The depicted example graph shows how information about the worker history is encoded. The node *History_Kowalski* represents the qualification-related data for worker *Kowalski* wrt. the product *DGC-XL* when assembling at line *U7*. The graph connects a model of the factory floor with a topology of manufactured products and the staff's profiles.

The worker model is implemented using the open-source, Java-based graph database Neo4j [7]. Querying of the model from within the application is realized by accessing Neo4j's RESTful interface. The model's contents are derived from a domain ontology (OWL) by automatically mapping the ontology's concepts and relations into the Neo4j graph database format (T-Box). Once created, the graph can be populated and updated at runtime with dynamically-changing data like a worker's history, retrieved by logging assembly operations.

There are several reasons for choosing a graph database above conventional storage formats like, e.g., SQL databases. Regarding performance for path operations (like in the above example) in highly-connected data, relational databases can become overburdened by queries of increasing complexity due to joins and index lookups; whereas in graph databases, which use index-free adjacency in traversing from node to node, query latency is relatively independent of the database size and the number of connections ([8], chapter 2). Note that denormalization for relational databases is not an alternative here, since the data model is not tailored exclusively to the needs of the staff planning app, but is instead meant to provide a flexible, multi-purpose source of semantic information, with diverse applications reading from and writing to the model. This implies that a) we cannot anticipate what relations will be queried most frequently at runtime, and b) we need to prepare for easy model update (e.g., based on sensor data), in order to keep the graph a realtime model of the staff data. Consequently, there is no use in optimizinig read access for specific relations at the cost of slower write access.

 ISBN: 978-1-61208-353-7

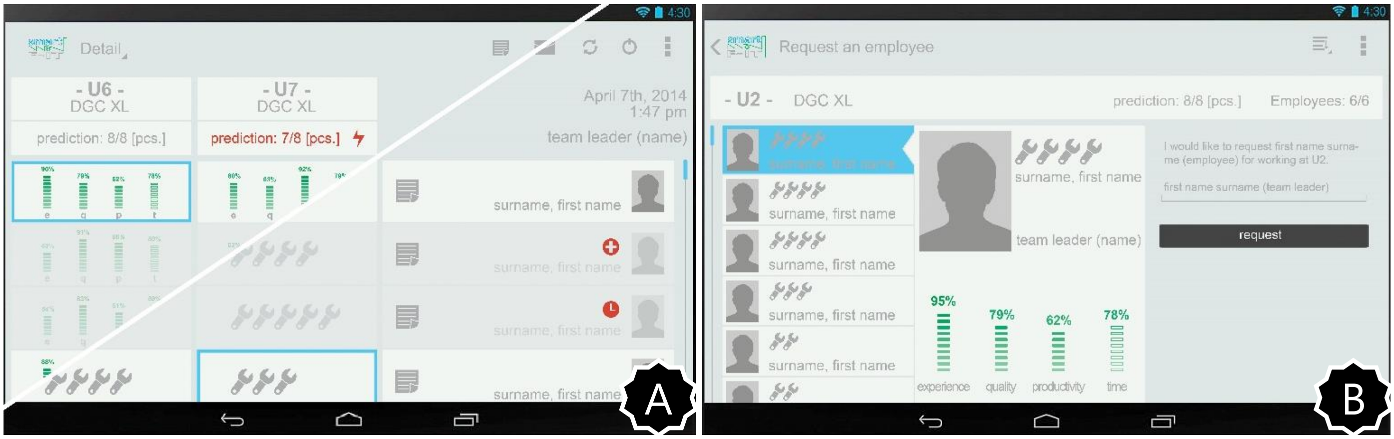

Figure 3. Mockups of the mobile application.

The most crucial benefit of graph databases in the context of Industry 4.0 is that they allow for an explicit, intuitive, and easily-expandable modeling of the complex semantic dependencies that exist in modern factories, and that the semantics of such graphs can easily be understood even by users unfamiliar with conventional modeling languages like UML.

VI. User Interface Design

A 7 inch tablet was identified to be most suitable for the daily deployment in a factory environment. It is small enough to fit in a team leader's pocket and offers enough space on the screen to present relevant information. In order to start the development of the screens, mockups of the mobile application were designed. The result is shown in Figure 3.

After a successful sign in, the home screen is presented (*screen A*). The screen visualizes the manufacturing lines and allocated workers the respective team leader is supervising. Events such as sick notes or delays are visualized by specific symbols. The absence of a worker can result in a deviation of the planned quantities for the day, which is visualized on top of the manufacturing line columns. To get an impression of the worker's fitness for the manufacturing process on a line, a 5 point ranking is visualized in form of wrenches.

If the team leader wants to break down this aggregation of worker skills, he or she can switch to a detailed view. The three qualification parameters *experience*, *quality*, and *productivity* (cf. Section IV-A) are visualized in form of bar graphs. As a fourth parameter, *time* indicates when the recommended dwell time limit on this line is reached. To trigger or execute a staff shift, the team leader changes to *screen B* by touching the matrix position he or she wants to edit on screen A. Screen B presents a list of suitable workers according to the weighted worker ranking. If the team leader selects a worker, his or her impact on the planned quantity numbers is visualized on top of the screen.

VII. Discussion and Outlook

In this work in progress, the concept of a mobile application for team leaders, to support them in making ad hoc planning decisions, was proposed. Recommendations of suitable staff shifts are based on a dynamic worker model that was implemented using a graph database. A Web service makes relevant information available that is presented on a mobile user interface.

In the next step, a prototype of the application is realized. The linking of relevant information sources forms the first challenge to allow a field test in the factory. Feedback of the team leaders will show if the desired improvements were achieved and will flow into enhancements of the prototype. Additionally, a useful data exchange between information sources, such as the factory worker model and the manufacturing line model, might result in new optimization strategies for an optimal worker-to-manufacturing-line allocation. On the other side, the information from the factory worker model might be used to provide workers motivating feedback on their daily work.

Acknowledgment

We would like to thank Nadja Rutsch for the design of the graphical user interface mockups presented in Figure 3. This research was funded in part by the German Federal Ministry of Education and Research (BMBF) under grant number 01IS13015 (project SmartF-IT). The responsibility for this publication lies with the authors.

References

[1] BMBF, "Project of the Future: Industry 4.0," 2014, URL: http://www.bmbf.de/en/19955.php [accessed: 2014-04-10].

[2] K. Ashton, "That 'internet of things' thing," 2009, URL: http://www.rfidjournal.com/articles/view?4986 [accessed: 2014-04-10].

[3] J. Kletti, Manufacturing Execution Systems (MES). Berlin; London: Springer, 2007.

[4] RWTH Aachen University IAW, "ENgAge4Pro Project," 2014, URL: http://www.iaw.rwth-aachen.de/index.php?article_id=20&projid=89&proalias=ENgAge4Pro [accessed: 2014-04-10].

[5] Fraunhofer IAO, "EPIK Project," 2014, URL: http://www.epik-projekt.de/ [accessed: 2014-04-10].

[6] ——, "KapaflexCy Project," 2014, URL: http://www.kapaflexcy.de/ [accessed: 2014-04-10].

[7] Neo Technology, Inc., "Neo4j," http://www.neo4j.org/, [Accessed 23-June-2014].

[8] I. Robinson, J. Webber, and E. Eifrem, Graph Databases. Beijing: O'Reilly, 2013. [Online]. Available: http://my.safaribooksonline.com/9781449356262