# Evolutionary Social Knowledge Graphs for Individual and Organizational Learning

Christoph Greven, Hendrik Thüs, Mohamed Amine Chatti, Ulrik Schroeder
Learning Technologies Research Group
RWTH Aachen University
Aachen, Germany
email: {greven, thues, chatti, schroeder}@cs.rwth-aachen.de

***Abstract*—Knowledge has been identified as one of the most important resources that grants organizations certain competitive advantages. Hence, while dealing with problems like the demographic change, organizations are trying to preserve the knowledge of their members. Its informal and network character, as well as its little half-life demand high standards to record, store and maintain the knowledge which needs to be consider for the development of an appropriate knowledge tool. This paper presents a novel knowledge management system which addresses the scattered, distributed, flexible and interrelated nature of knowledge. It therefore brings different important aspects together and utilizes the personal knowledge management of individuals in working processes together with social collaboration methods for a global organizational learning process. The three reflection layers promote a continuous feedback loop resulting in high quality knowledge maturing. The paper mainly concentrates on the essential underlying architecture and knowledge structure that only makes the learning process possible.**

***Keywords-Knowledge Management System; Informal Knowledge; Network Knowledge; Knowledge Graph; Knowledge Evolution; Organizational Learning; Reflection.***

## I. INTRODUCTION

Since time immemorial, society tries to impart new knowledge to their posterity and therefore finds appropriate ways to capture it. As, over time, the kind and form of knowledge changed, the methods and techniques to store it changed accordingly. Knowledge itself is a controversial topic and scientists have many different ideas and views on it. But when it comes to a concrete software tool to deal with it, knowledge needs to be stored somehow. That is where traditional knowledge management systems fail because they are not able to map the complex characteristics that knowledge nowadays has. They are not considering the multimodality, flexibility, interrelation, and short life. As a consequence, a lot of potential remains unused or even gets lost.

The lost capacities are also sensed by the economy. While resource management in general has always been an important issue for the efficiency, companies start to realize that, beside traditional sources like money or workforce, expertise or knowledge is a new resource and good that drastically impacts their competitiveness. Having employees with a lot of expertise means an enormous advantage on the market. Hence, it is not astonishing that companies start to manage their knowledge like all other resources to keep, maintain, and expand their knowledge.

There is a common consensus on the significance of knowledge for economic success. However, its extraction and collection is not as easy. Organizations as well as society have to face different recent problems: the running demographic change, an enormous information overflow, overspecialization in special fields instead of heaving general problem-solving competency, the high percentage of tacit knowledge which is not tangible from outside, little exchange of experience which is often related to missing communication possibilities, a lack of knowledge application after trainings or workshops, no feedback for the authors from the real practitioners, little reusability of knowledge due to the rigid old structures, very slow publication procedures of new knowledge, for example owed to only annually meeting committees, or very specialized and often mobile workplaces that require considerably different demands.

Regarding the learning or knowledge management process, there are different challenging phases. First, the knowledge is in the employees’ minds. The largest share is not factual but tacit knowledge. Thus, it is difficult to make this knowledge explicit and verbalize or formalize it. Second, employees that have certain competency do not communicate and share it. The most valuable knowledge is useless if anyone can access it. Last, the knowledge needs to be kept up to date and be adapted regarding the latest changes. A quality assurance of the individual knowledge is mandatory to eliminate errors and misconceptions.

All the present problems can be addressed by aspects of knowledge management systems. Therefore, this paper presents a novel learning management system that owns the potential to help out of misery. Thereby it presents the overall process of knowledge management but clearly focusses on the underlying unique architecture and structure which makes this kind of interactions possible.

The remainder of this paper is structured as follows: Section 2 gives a brief overview about recent knowledge management approaches. Section 3 introduces and explains the knowledge management process and hence the individual as well as organizational learning. The concrete underlying structure which defines how knowledge is finally represented and treated is depicted in Section 4. As this is the major part of this paper, different aspects like knowledge entities, relational behavior, versioning or access management are described in more detail. Section 5 gives some information regarding evaluation efforts. Finally, Section 6 concludes the approach so far and gives an outlook of future procedures and possible extensions.

## II. Knowledge and Learning

Organizations had high hopes in knowledge management when it came up in the 90s. The topic knowledge management has been discussed a lot in literature and there are various different opinions on it. Earlier discussions are about whether to understand knowledge as a thing [1][2][3] that can be stored like in simple information management. Others see it as a standardized process that learners always pass through but which is not really flexible. One example is the well-known SECI model [4] which seems to be flexible at a first glance but which is not indeed. More recent approaches are concentrating on the knowledge worker himself and realize knowledge as well as learning as highly personal. Personal knowledge management puts the learner and his tacit, implicit knowledge in focus [5][6][7]. In contrast to earlier understanding of knowledge management, this is steered by the user and hence following a bottom-up instead of a traditional top-down approach. Still, the nature of today’s knowledge cannot be mapped.

Today, organizational structures like companies invest enormous amounts of money in education and training. Still, those efforts cannot prevent regular incidents with e.g. breakdowns appearing in the public media every now and then. Many of those effects are owed to omissions of knowledge management and a corresponding quality assurance. Especially traditional organizations are only slowly adapting their management strategies and often count on outdated approaches such as simple learning content or asset management systems. Although, such systems got more interactive with the new possibilities of the Web 2.0 focusing on the worker a lot more, platforms like wikis are still struggling in the professional working environments. They cause additional expenses and the active involvement of people is hard. Personal learning environments try to address the individuality of the learners and their tacit knowledge, but in most cases they are only a set of enforced tools that do not fit in naturally. Further aspects like workplace mobility make knowledge management even more difficult while the new powerful mobile devices offer a lot of still unused potential. Systems for mobile distribution of digital documents [8] or question and answering systems [9] show that companies are trying to find a way out, but that the overall transfer and integration of new approaches is rather slow. And nevertheless, not all current problems like the short half-life, dispersion and fragmentation or interrelation of knowledge are addressed.

The approach presented in this paper is based on the Learning as a Network theory [10] which unites the concepts of network learning, complexity theory, and double loop learning. It regards learning and working as one thing and addresses the challenges described.

## III. Evolutionary Knowledge Progession

The idea behind the whole concept is a 3-layered knowledge reflection process which promotes personal learning, naturally leading to organization learning as well. As depicted in Figure 1, the process concentrates on the single knowledge worker and his knowledge. As a certain knowledge maturity or quality has been reached in one phase, it can be raised to the next level. At the same time, experiences on higher levels always reproduce a knowledge flow backwards.

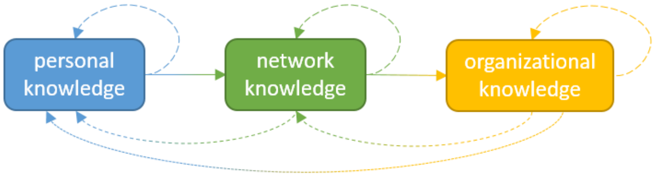

Figure 1. Three-layered knowledge reflection loop.

The first phase starts with the personal knowledge management and learning process of each individual. The individual refers to official material like instruction rules, documentations, guidelines, or trainings in his everyday working process. While using it in his every day work, he gains experience and has the possibility to create his very own multimedia notes and aids. With the help of his newly gained knowledge, he can solve similar problem situations in future. By and by, he changes, enhances and corrects his thoughts in this feedback loop which results in high quality working aids. At any time, the individual can decide to exchange his knowledge with a selected group of peers allowing them to participate in his experience.

In the second phase, knowledge is communicated amongst the personal knowledge networks of the individuals. Discussions come about and argue on certain approaches, understanding or best practices that have been experienced by the individuals. Besides the open dialog, users also have the possibility to rate all kinds of material available, from official guidelines to answers to comments. The new insights gained from the discourse influence the knowledge in the network as well as the personal opinions so that parts are revised resulting in a higher quality.

In the third and last phase, the created knowledge should flow back to the original authors so that it can be didactically reworked and integrated in the official organizational knowledge. As new versions, it can be published for all the workforce again where it represents the basis for future working processes. The overall continuous feedback loop starts again. Intelligent methods help editors to find and discover problem areas or highly valuable knowledge in the system.

In principle, the presented process is generic and can be applied in various kinds of scenarios. The related project – Professional Reflective Mobile Personal Learning Environments (PRiME) – in which this model has been developed, concentrates on mobile field services that profit the most [11]. An according architecture of mainly mobile clients of visualizers and manipulators communicate with a central knowledge repository through service interfaces. The mobile application ecology thereby supports the personalization of each individual and can be flexibly adapted to the current needs of the worker [12][13]. The presented process appears to be unspectacular, but to really establish such a continuous feedback loop, many different aspects need to be considered. The next section explains the underlying complex knowledge structure step by step, for without no system would have been possible.

## IV. REALIZATION

This section gives more details on the concrete realization and implementation which is the foundation for the before mentioned knowledge process. Therefore, the different knowledge entities are explained, relationships are introduced, the versioning is described, annotations and ratings are pointed out, and finally the authorization system is explained. The paper does not restrict to a fixed technological implementation as a selection of such heavily depends on the application scenario and environment.

### A. Entities

In whichever way knowledge is generated or understood, when it comes to a concrete implementation it has to be stored in a data base somehow. The system supports the separation of concerns and concentrates on the content and its structure and not on its visual representation. The own knowledge structure is format-less and does not contain general style information. Figure 2 shows a very simplified class diagram of the elements and relations that are introduced in the following sections. There are two major entities which represent different aspects of the knowledge.

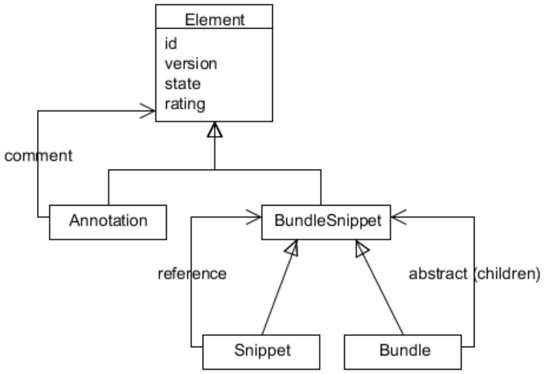

Figure 2. Simplified CD of knowledge structure.

#### 1) Snippets

Snippets are multimedia-based atomic units of knowledge. A single snippet might cover a paragraph, an image, a short video, an engineering detail drawing, a 3D-model, and so forth. It is self-contained and from a semantic point of view it would not make sense to split it up any further. Snippets are reduced to content and do not address possible visual presentation. Hence, they abdicate the most formatting and style information and are restricted to those that contribute to the actual content. That are, e.g., structural information like listings or tables or style information like bold or underline. Beside the content, snippets hold further meta information like an abstracting title, information about the author, etc. Summarizing, they represent a smallest semantic unit of concrete, directly applicable content.

#### 2) Bundles

Bundles are logical, semantic units that are thought to group an order of knowledge entities such as snippets, or bundles themselves. That means it is a recursive data structure which allows to set up several layers of bundles and snippets. It can be understood as a tree where the inner nodes are bundles and the leaves are snippets whereas leaves are allowed on every level. Bundles do not contain content themselves but such as snippets they hold some additional meta information, e.g., a title. They can be compared to chapters or sections in a book, where text (snippets) may be placed on each level as, e.g., an introduction of the section, or inside a section. Bundles are logical groupings of content and form an enclosed object themselves regardless of the surrounding content and where they are to be found. Summarizing one could say that bundles mainly realize a named list of references to sub-elements.

### B. Relations

The previous sections introduced snippets as content holders and bundles as semantic structures. Without any relational associations they were lose, incoherent pieces without any use. And as already mentioned, the most important aspect of today’s knowledge is still missing: interrelations. That is why the system offers different kinds of relationships between elements that all have a special semantic. Simplified, one can first think of bidirectional associations (see next section for more details). The first and most important relation has been implied before already. It is the parent-child relation of bundles (parent) and bundles or snippets (children). The recursive component allows a hierarchical representation of knowledge in a tree-like structure. Through the relation, bundles are able to somehow abstract and aggregate their subtree and make it usable as a whole. Official documents or training exercises are arranged the same way: there is a general topic which is split up in subtopics, and so forth. Although traditional documents have a tree structure (also see Section 4.H), once they are in the system they (or parts of them) can be reused in different contexts. That means they can be embedded via the parent-child relation in other bundles resulting in potentially more than one parent for each bundle and hence many simultaneous interwoven trees.

Oftentimes, knowledge refers to some other knowledge. As an example, specifications often contain phrases like "see chapter x" or "as in figure y". There is a corresponding relation – namely references – which allows snippets to cross-ling additional elements which do not even need to be in the same tree. As a result, the trees-structure becomes a real graph. Additionally, there are further relations which are of minor importance. For example, there is a "based on" association telling that this element or its version is based on content in some other snippet, bundle or comment (see Section 4.B for more details). In particular, this can be used to "thank" a user for his contribution.

### C. *Versioning and States*

For a knowledge management system that also offers official material like documentations, instruction rules, guidelines, etc. it is important to keep track of the changes that have been done in the system. That is, e.g., related to the responsibility of the authors. Hence, a complex versioning system has been created which covers the system's entities and their relations.

Elements, i.e., bundles and snippets, are uniquely defined by their id and their version number. Newly created elements get a novel id and start with version 0. As soon as new versions are created, new physical elements are injected, having the same id but an increased version number. This way, the history of one single element can be traced by showing all elements with the same id and all the different versions. Essentially, a new snippet version means a changed content (or meta information like its title). It is to be created when the knowledge atom needs to be updated, changed or enhanced. In contrast, a new bundle version means a changed structure (or its meta information). As a bundle only defines its children and hence the subtree in the hierarchy, adapting it means adding, ordering, or even removing a child element.

The whole knowledge structure is based on the directive that the newest version of an element is understood to always be the best version of it. At least in relation to the current state of knowledge in the system. That has big implications on the whole process and structure. As stated before, knowledge is not represented redundantly. There are no copies of elements but multiple usage of an element is realized by multiple relations to one single element. Considering the "the newer the better" rule it would be comfortable if relations always addressed the newest versions of elements as all previous versions are thought to describe the same issue but in a less optimal way. That is how the system has been implemented. This implies, that as soon as an element is changed into a new version, all other elements which have relations to the changed element are now directly referring to the new version. As an example, one can think of an exploded view of a machinery which is used in a book's chapter and in a workshop presentation. In case the image needs to be corrected, the new version is immediately included in the book and presentation as well. The versioning together with the referencing keeps the knowledge structure very flexible and adaptable.

The authoring process of creating new versions commonly includes a phase, where elements are under construction but not yet finished to be accessible for everyone. That is not covert in the system so far. To satisfy this requirement, elements are extended via states. The idea behind states is to describe in which maturity phase an element is. When a new version is created due to changes of the current element, the version number is incremented. Furthermore, the element is *working* state then. All temporal independent changes of the element are directly applied and do not result into a new version until the author has finished his work. And yet it is no different if changes are done within a minute or over several weeks. The working state also has some other influences. For example, working versions of elements cannot be found by someone who does not own authoring rights. The element remains in working state until it is actively published by an author. That means the version number stays the same while the state is changed to *published*. At this point of time, the element reached an official character of quality and is accessible for the target group. From then on it is not possible to adapt the element, and changes result in new working versions with incremented version numbers respectively. Besides the *working* and *published*, there are other states, for example *initial* for automatically imported elements with additional restrictions. Figure 3 shows the different versions/states of an element and the possible transitions.

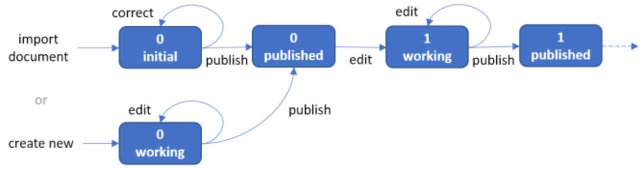

Figure 3. Knowledge versioning process.

The versioning of the bidirectional relations and the different states do not mesh with each other very well. The structure of elements and relations is thought to have one best element for each knowledge issue. The concept of non-published new versions leads to a violation as there is a coexistence of two elements describing the same things at a time. Of course, it would not make sense to show both of them in the tree or link the new version already as it has not been finalized yet. To address this challenge, it has been decided to slit the bidirectional relations up into two unidirectional relations, which are not synchronized at the same time but with a delay of the different state transitions.

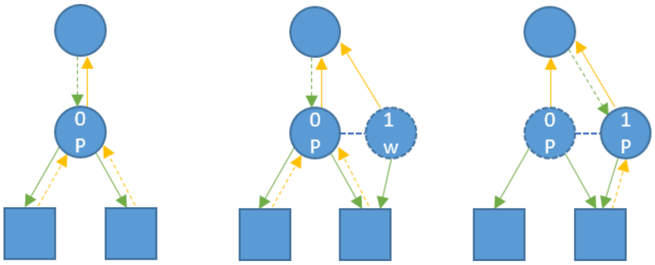

Figure 4. Evolving Knowledge Graph.
Left: published bundle (version 0) with two children and one parent bundle.
Middle: one snippet is deleted in working version 1 of bundle.
Right: updated bundle (version 1) is published.

Figure 4 shows an example of one root bundle, containing one other bundle, containing two snippets itself. One can see that the return paths of the middle bundle's relations (dashed) are only set the moment the element is published. When being in working state, it only adapts or rather creates the relation parts of which it is the owner (solid). The figure also shows how the child relation (green) of the top bundle is altered on publish. Splitting up the bidirectional relations added some complexity as there two independent structures by the going and returning part of the relations. But since the modification process of the relations follows strict rules, it is a good tradeoff.

### D. *Annotation System*

At the beginning of this section, the two main entities snippets and bundles have been introduced. Actually, there is one more element which has a deep impact on the utilization of the whole system. In contrast to snippets and bundles, which represent something official or thought trough in the system, annotations are personal notes of individual users. Just like snippets, they are multimedia objects. The creation of annotations via certain tools is very easy so that users can ad-hoc record audio, take a photo, create a video, list some issues, create a sketch, etc. Thus, it is easy to grab non-formal situational knowledge. Furthermore, a short description can be added explaining the note in some detail. The real profit is earned when annotations are stuck at any other knowledge element in the system. I.e., with the help of annotations the user is able to create his own working aids and extend the global knowledge with his own experience. For the moment, his annotations are private and can be seen by himself only. Nevertheless, it seems as if they are part of the available knowledge graph. Whenever the user works with the available material, from now on he directly receives his embedded annotations as additional help. As soon as the aids are believed to be valuable enough, the user has the possibility to share them with a self-determined group of other users (see sections 4.F and 4.G). Annotations can even be used to comment on already available annotations of other users encouraging discourses and allowing the author of the knowledge element to receive useful feedback and insights. That means the system avails communication and exchange of knowledge where it was not possible before, e.g., due to a job profile like field services. On the one hand, helpful contributions can be used to improve the knowledge in future versions. On the other hand, the author realizes misconceptions he would not have known otherwise and has the chance to respond to them. From an architectural point of view, the annotation structure is infinitive, but due to usability the level of annotations is limited to 2 levels via the program logic. That way, answers to annotations on elements are still possible but discourses do not get too complex and unclear. As already mentioned, annotations are similar to snippets although they serve a different purpose. The likeness can be used in such a way that an element's author can use annotations as templates for new elements or improved versions. That way he can easily embed, e.g., a photography of a mechanic taken of a machine. The mechanic himself gets involved in the process and can identify with the new material due to his input.

### E. *Rating System*

Besides annotations, knowledge workers have another possibility to interact with the available knowledge. An extra rating system has been integrated into the structure which allows users to rate all kinds of elements, i.e., snippets, bundles and annotations. While ratings on annotations and snippets refer to the content or the remark, a rating of a bundle expresses the quality of the compilation. For example, that includes which subsections a chapter has, how subsections are ordered, whether the collection of elements is semantically complete, etc. Due to the rating system, it is very easy for users to communicate their thoughts without too much effort. Still, the input can be used to advice high quality knowledge and identify problem areas to contribute to an overall quality assurance. As users do not vote too much and if they do they do not tend to down-vote, the graphical user interface needs to assure a quick access of a simple rating mechanism like, e.g., positive stars. This is also very important for the authors and all users in the system. The awareness of activities in the system has a motivating effect on its participants and their willingness to contribute.

### F. *Diverse Group System*

Annotations and other elements can be share with selected peers. To achieve this and to simplify the communication and distribution of material, a diverse group system has been created. Groups are collections of peers in the system and designed in a rather generic way. Via different characteristics, such as visibility or admission procedures, it is possible to easily create various different kinds of groups. It is feasible to have personal unidirectional friend lists which are only visible for the user himself, circles like in google+, groups which are used for commonalities such as working locations or occupational profiles, more formal groups that may represent a successfully passed training, groups that reproduce department structures, and so forth. Table 1 shows some possible types of groups regarding some of their features. They are used for communication purposes as well as access management of knowledge elements in the system as described next.

TABLE I. GROUPS AND THEIR FEATURES

| Feature \ Group | Personal Friends List | Group of Colleagues | Official Department | Working Location |
|---|---|---|---|---|
| Open | No | No | No | Yes |
| Visible for all | No | No | Yes | Yes |
| Applicable | No | Yes | No | No |
| Invitable | No | Yes | No | Yes |

### G. *Access Rights Management*

Not only from an organizational point of view it makes sense to restrict the users' access to selected content. This is reasonable if there is, e.g., security-related material for which's use the worker is not educated. The presented group system is an optimal basis for the needed rights management. In the knowledge system, the creator is in full control of his knowledge and has the power to grand additional rights. As there is no one system-wide privileged author, it depends on the element and the current situation whether some user owns this authoring role or not. Access rights are related to single elements and hence for every snippet or bundle – at least in principle – it could be different. Of course, the user interfaces simplify the process such that authors can adapt the access for a whole (tree) structure at once. Besides the special *creator* role which allows to retain full power over a created element, there are several other roles. There is a right to *read* and show elements, and a right to *write* and change an element. Access rights cannot only be assigned to a single user but also to groups. That way it is very easy to allow a certain group of people to use some material, e.g., the handouts of a workshop. As soon as more people pass the corresponding training, they only need to be invited to the group which is already authorized. Another *manage* right allows users to allocate rights to other users and pass the power. For example, his makes sense if there is committee that is responsible for some knowledge. One last special right is called *REFERER*. The owner of this right is able to only add read access to others. That becomes important if authors want to include elements of other authors into their bundles. Commonly, they would not have power about the rights allocation of the included elements. To still be able to offer reading rights to their audience, they can get the REFERER right which enables them to add readers to foreign material. Table 2 summarizes all different rights again.

TABLE II. ACCESS RIGHTS

| **Access Right** | **Declaration** |
|---|---|
| CREATOR | Implies all the access rights and cannot be revoked |
| READ | Find and read-out element |
| WRITE | Change element |
| MANAGE | Grant and revoke other rights |
| REFERER | grant READ rights |

### H. *Linkage of Traditional File Formats*

The cold-start problem is a negative effect which many new software systems suffer from. That occurs, when there is too little data in the system to really use it effectively. However, employees need their material and cannot trust in a system that only covers some aspects. The risk of falling back into old habits and not accepting the system as a whole is too high. Also, organizations already have an enormous mass of material in digital file formats of traditional tools like MS Word, MS PowerPoint, and others. Hence, it would be a tremendous help to transfer those into the new system structure.

Different importers have been created that take, e.g., a MS Word's docx file and convert it into the system's internal structure of snippets, bundles, etc. The modules analyze the original document and utilize different kinds of information to build up the hierarchy. For example, the different levels of headlines can be used to recognize the different bundle levels, or paragraphs can be used to determine text units for snippets. This automation disburdens the human authors a lot. As it is not possible to create an automatism for all theoretically possible inputs, the modules only generate suggestions which are stored via a special initial state. Authors then have the chance to correct the material in regard to inaccurately detected elements. That means combining snippets that should have been recognized as one, splitting up material in further units, and so forth. After publishing the new structures, they are ready to be used like structures that have been created from scratch in the system. Elements in the system are neither restricted to their former visualization nor to their previous format.

Summarizing, the presented knowledge structure represents a network of knowledge elements (snippets, bundles, annotations) and their interrelations. Version control realizes a constant graph evolution and still allows to reproduce the history of knowledge. Social aspects like annotations and rating together with the group system are the basis for lively exchange of knowledge between peers. Via the rights and roles management, it is very easy to grant access to different users. To overcome the start-up problems, import modules are able to automatically transform traditional documents into the new knowledge structure.

## V. EVALUATION

The presented approach has been developed in connection with a project named Professional Reflective Mobile Personal Learning Environments (PRiME). It is a joint research project of the Learning Technologies Research Group from RWTH Aachen University and DB Training, Learning & Consulting from Deutsche Bahn AG. It is sponsored by the Federal Ministry of Education and Research via the German Aerospace Center. Although the system developed in PRiME is designed in a generic way to fit many different scenarios, the most benefitting job profiles are mobile field services. As proof of concept, we address a first group of mobile mechanics from the car inspection service of the long-distance passenger transport DB Fernverkehr AG, as well as related trainers, training developers and specialist author. The mostly qualitative evaluations in form of interviews and work tasks show broad acknowledgement and positive feedback from all involved roles. It also emerged, that in principal the current traditional processes are similar to the proposed ones but so far very uncomfortable and slow in comparison to the new possibilities in the PRiME system. Current employees make their own way to cope with the addressed problems and look for workarounds or their very own tricks anyway. Thus, the need for an organization-wide uniform solution is clearly there.

Further quantitative evaluations with, e.g., questionnaires in combination with some broader field studies are running at the moment. First figures adumbrate that the implemented system and its underlying model can really embed in the everyday work life and naturally support the workers in their working process. The results will be published accordingly.

## VI. CONCLUSION AND FUTURE WORK

Knowledge has been identified as a very important resource that greatly impacts the competitiveness. Hence, there are endeavors to improve the collection, distribution and the enhancement of knowledge and manage it like any other good.

In this paper, we introduced a promising approach that unites different – so far independent – aspects like complexity, network character, social aspects, and quality assurance together into one process and model to naturally fit in and characterize the real knowledge process. The underlying structure has been explained in more detail and allows the distribution of official knowledge, the record of tacit knowledge, sharing it with peers, and improving it due to an annotation and rating system. The whole system results in a three-layered learning process starting with the personal individual, over his social network to the whole organization. This continuous process of knowledge evolution and maturation has some additional benefits:

- Knowledge does not leave the organization with its members and hence does not need to be reproduced again and again.
- The concept promotes a mentality to work together and benefit from one another instead of destructive competition.
- By their contribution, knowledge workers feel involved and can identify easily with the organization.
- The organization's focus is more on its employees and their abilities.
- Due to their participations, users enjoy respect and appreciation.
- Users have the chance to communicate and experience social aspects that have not been possible before.
- Very high reusability of knowledge by virtue of the atomic elements and their linkage.
- Faster publication procedures of official knowledge as authorities can concentrate on point by point enhancements.
- Authors get to know about misconceptions and can reveal them.
- Authors can fall back on an enormous collective know-how and quality assurance a lot faster due to the high involvement.

The presented knowledge system is already able to map the whole knowledge process from individual to organizational learning. Nevertheless, there are many aspects that can and should be enhanced. The introduced structure can cope with the spread and cross-linked character of the knowledge. Still, one idea is to add more semantic meaning to elements by, e.g., introducing categories or types of knowledge units to improve its discovery and offer it more selective. It has to be further researched if it is possible to deduce concepts or taxa from the structural content in different areas of application. Once formalized, ontologies can then abet improvements in, e.g., identifying situationally important knowledge.

Content can be created from scratch or traditional file formats can be imported by specific modules, as described before. Even if the whole management and usage process is represented in the system, there are still situations where other formats outside of PRiME are needed. That could be a PowerPoint presentation in a workshop or a Word document for external companies that do not have access to the system. Analog to the import modules, export modules will be able to generate traditional documents back from the special systems own structure.

The current employees are used to their present toolset which is in most companies the Microsoft Office Suite. Instead of forcing them to use yet another system, there are approaches to develop assistance systems that integrate PRiME into their common working environment. As an example, plugins for their text editors can help to stick with some guidelines to gain the maximum profit and less rework from the import modules.

Learning Analytics is a powerful tool that allows better learning data analyses. On the one hand, it can improve the automated feedback for authors so that they see points of failure or misconceptions at a glance. On the other hand, it can optimize the user's handling of material in his working process. Further collection of context data [14] like time, location, situation, etc. can help to offer the right knowledge which is required for the individual in his current unique situation. Ideally, it could dispense with former traditional searching.

Aside the mentioned aspects, there are many more possible extensions or improvements which could also cover topics like assessment. It remains to be seen how the system will be accepted in long-term studies and if there are other aspects of higher priority such as the improvement of user motivation.

## REFERENCES

[1] P. Hildreth and C. Kimble, "The duality of knowledge," Information Research, Vol. 8 No. 1. 2002.

[2] C. Kimble, Paul Hildreth and Peter Wright, "Communities of practice: going virtual," in Y. Malhotra (Ed.), Knowledge Management and Business Model Innovation, Idea Group Publishing, Hershey (USA)/London (UK), pp. 220–234. 2001.

[3] Y. Malhotra, "Integrating knowledge management technologies in organizational business processes: Getting real time enterprises to deliver real business performance," in Journal of Knowledge Management, Vol. 9 No. 1, 2005, pp. 7–28.

[4] I. Nonaka and H. Takeuchi, "The Knowledge-Creating Company: How Japanese Companies Create the Dynamics of Innovation," Oxford University, New York. 1995.

[5] G. E. Gorman and D. J. Pauleen, "The Nature and Value of Personal Knowledge Management," in D. J. Pauleen and G. E. Gorman (Eds.), Personal Knowledge Management: Individual, Organizational and Social Perspectives. Gower Publishing Limited, Farnham Surrey, England, pp. 1-16. 2011.

[6] L. Prusak and J. Cranefield, "Managing your own Knowledge: A Personal Perspective", in D. J. Pauleen and G. E. Gorman (Eds.), Personal Knowledge Management: Individual, Organizational and Social Perspectives. Gower Publishing Limited, Farnham Surrey, England, pp. 99-114. 2011.

[7] D. Snowden and D. J. Pauleen, "Knowledge Management and the Individual: It's Nothing Personal", in D. J. Pauleen and G. E. Gorman (Eds.), Personal Knowledge Management: Individual, Organizational and Social Perspectives. Gower Publishing Limited, Farnham Surrey, England, pp. 115-128. 2011.

[8] "DB Regio und DB Training gewinnen Deutschen Bildungspreis 2013“, https://www.db-training.de/dbtraining-

de/start/news1/6012202/bildungspreis2013.html. reviewed: March, 2016.

[9] J. Finken, D. Krannich, and B. Tannert, “Kodin-Kfz - A Collaborative Diagnosis Network,” in Proceedings of ICL 15th International Conference on Interactive Collaborative Learning. Sept. 2012, Villach, Austria. IEEE Conference Publications. 2012, pp. 1–3.

[10] M. A. Chatti, U. Schroeder, and M. Jarke, “LaaN: Convergence of Knowledge Management and Technology-Enhanced Learning,” in Learning Technologies, Vol.5 No.2. 2012, pp.177-189.

[11] C. Greven, M. A. Chatti, H. Thüs, and U. Schroeder, “Context-Aware Mobile Professional Learning in PRiME,” in “Mobile as Mainstream - Towards Future Challenges in Mobile Learning : 13th World Conference on Mobile and Contextual Learning”. Springer. 2014, pp. 287-299.

[12] C. Greven and U. Schroeder, “²SIMPLE: Symbiotic Interrelated Seamless Integrated Mobile Personalizable Learning Environments,” in Proceedings of DeLFI Workshops 2015. 2015, pp 232-243

[13] C. Greven et al., “Seamless Application Ecologies as Mobile Personal Learning Environments,” in “DeLFI 2015 – Die 13. E-Learning Fachtagung Informatik der Gesellschaft für Informatik e.V”. 2015, pp. 57-69.

[14] Thüs et al., “Mobile learning in context,” in Int. J. Technology Enhanced Learning, Vol. 4, Nos. 5/6, 2012, pp. 332-344.