Scrutinising Competencies: Retraceable Clouds of Learning Goals in the APOSDLE System

Victor Manuel García-Barrios[1], Günter Beham[2], Barbara Kump[3]

[1] Institute for Information Systems and Computer Media, Graz University of Technology, Inffeldgasse 16c, 8010 Graz, Austria
vgarcia@iicm.edu

[2] Know-Center,
[3] Knowledge Management Institute, Graz University of Technology, Inffeldgasse 21a/II, 8010 Graz, Austria
gbeham@know-center.at, bkump@TUGraz.at

Abstract. The APOSDLE research project aims at developing an integrated framework of tools to support work-place learning. The resulting system will assist workers during their tasks by connecting and optimising their working, learning and collaboration activities. For this purpose, the system needs complex models for the representation of workflows, competencies, knowledge domains as well as user profiles and interactions, among others. In that context, this paper focuses on the critical aspects that arise from conveying to system users how their individual task history, competence-based learning goals and learning activities are interrelated. As a result, a solution approach is presented, which places learning goals at the centre of a user profile visualisation tool. The solution approach follows the principles of dynamic lists and tag-clouds in order to improve the scrutability of individual user profiles and to overcome the difficulty of conveying in a human-readable form the usage of complex models.

Keywords: Work-integrated Learning, Cloud-based Visualisation, Learning Goals, Competence Management, Scrutability, User Profiling.

1 Introduction

In general terms, the development of personalisation-pertinent systems is conducted under the premise of *one size does not fit all*. The end-users of such systems are aware of the increasing amount of well-tailored information they may access for their particular needs or goals. Moreover, they are aware of the fact that in the majority of cases, they have to pay the price or hazard the consequences of delivering personal data to ensure those services. According to the results of several surveys, most of the users are willing to do so; e.g. as stated in [1], 76% of the survey subjects expressed strong interest in receiving personalised content, and 45% were more likely to visit Web sites that provide personalised recommendations than sites without them. The high relevance of these systems can be identified in various application areas, such as e-commerce, recommender systems and adaptive e-learning ([2] [3]). Thus, rather than a trend, there exists a need for personalisation-pertinent systems in distinct situations of modern life [4], e.g. in work-place learning in order to increase or optimise the individual competence level of corporate workers. But apart from the fact that personalisation continuously gains interest within the research and end-user communities, the success and efficiency of a personalised service depends strongly on its technological implementation. The core components of such systems have to deal with high accuracy when assuming to *know their users' interests and goals*, i.e. the applied reasoning methods, the needed model representations as well as the acquired (or inferred) information in the individual user profiles are critical issues from the point of view of system developers.

Against this background, one of the biggest challenges met by developers of personalisation-pertinent systems is making it easy comprehensible to users *how and why the system has delivered a personalised service* as well as *which information in the individual user profile has been used*, but at the same time, *hiding the level of computational complexity behind the user interfaces*. Thus, the information in the user individual profiles should be easily scrutable [5]. In general terms, the difficulty of conveying to end-users the structure, state, meaning and usage of an abstract model increases with e.g. the complexity of its internal representation and computation, the degree of its evolution, the lack of its human-readable descriptors, as well as the loss of the usability of its visualisation. Within the context of personalised services of *work-integrated learning* systems, this paper presents a visualisation solution approach that deals with the high and dynamic complexity of competence-related models in the first prototype solutions of the APOSDLE system.

The work and ideas presented in this paper are the outcome of the APOSDLE research project (Advanced Process-Oriented Self-Directed Learning Environment). As stated in [6], the APOSDLE system offers personalised learning support to their users while working with existing corporate-associated information and contributing with new information to the corporate knowledge repository. Within the scope of the research project, these persons are called *knowledge workers* and include engineers, researchers, software developers, consultants and designers. The APOSDLE project follows a *Learn@Work* approach, i.e. learning takes place in the immediate working environment and context of the system user. In contrast with traditional e-learning systems, the system offers integrated support for the three roles of a knowledge worker at work-place: worker, learner and expert.

The topics driving the reminder of this paper can be summarised as follows. The practical scope of the APOSDLE project is defined by the integrated support of the main activities of knowledge workers: *working*, *learning* and *collaborating*. Thus, the (semi)automatic identification of their task-based learning goals plays a central role in the system. In order to fill a competence gap in a working situation, the APOSDLE system personalises its services for the user and recommends documents, learning events or collaboration possibilities with experts. Due to the complexity of the models behind this functionality and taking into account that the users of the system need a view on the personalisation-pertinent behaviour of the system, a simple and efficient visualisation of the relationships among the interacting models is required. Chapter 3 deals with this problem and gives an overview on the proposed solution approach. The paper concludes making references to related work as well as underlining open issues and future work.

2 Working, Learning, Collaborating

This chapter gives an overview on the research ideas and goals of the APOSDLE project. It introduces into relevant terminology as well as presents the general aspects of the overall system architecture and of its first prototype implementation (section 2.1). The focus in this chapter is set on the notion of a learning goal. An overview on the competence-based model of learning goals is introduced, and based on that, main requirements for the visualisation of an *individual chronology of applied & acquired learning goals* are defined. This central point of attention is described in section 2.2.

2.1 Process Oriented Self-Directed Learning

Within the scope of the APOSDLE project, and according to [8], the activities of a knowledge worker are mainly defined by overall goals and expected results instead of predefined task procedures. Thus, a knowledge worker may organise the structure of her activities with certain autonomy in terms of their timing and sequencing. As a consequence, a knowledge worker may switch to different tasks or domains in her workflows. This switching reflects the dynamics in a so called *user context*, wherein a knowledge worker switches to different roles in her working situations (e.g. from worker to learner, or from learner to expert).

In APOSDLE, the notion of *learning* refers to the advancement of knowledge and skills of knowledge workers, and includes the following characteristics: (a) work-place learning is integrated in the current working tasks of knowledge workers and utilises existing resources; (b) work-place learning activities aim at enhancing the performance of working tasks; (c) from the point of view of knowledge workers, work-place learning may occur spontaneously or unintentionally; (d) the learning needs and goals of APOSDLE users are derived from the tasks they currently perform; (e) learning activities emerge either from making use of available knowledge sources or in the creation of new knowledge (e.g. during collaboration events); (f) the results of learning activities (i.e. acquired knowledge and skills) may be directly transferable to the worker's working situation. [8]

The general system architecture of APOSDLE's first prototype, shown in figure 1 (top side), depicts its central software parts and their interrelations (*APOSDLE Tools*, *APOSDLE Platform* and *Backend Systems*). The architecture is based on a client-server software system, whereby the implementation follows a SOA paradigm (Service-Oriented Architecture) [6]. The Tools provide an interface to the user of the System, while the Platform provides the server-side functionality of the System. The bottom side of figure 1 illustrates the two sets of tools at disposal: the *Modelling Tools* (for experts to create formal models of user environments) and the *Workplace Tools* (for knowledge workers to use during work-integrated learning). Thus, the APOSDLE Tools represent the degree of operational complexity of the system. The computational (or functional) complexity of the system is given by the APOSDLE Platform (see [6] for details), which is mainly in charge of providing (a) foundation functionality to the whole System (Tools and Platform) through its Classification Service, Homogenous Access component, Semantic Service and Structure Repository Manager, and (b) a way of recommending resources for work-integrated learning through its Associative Network component and User Profile Service. The latter issue is based on the context of the learner, thus, on the one hand the Associative Network searches and retrieves context-dependent learning sources, and on the other hand the User Profile Service manages fine-grained information about the learners and their individual contexts.

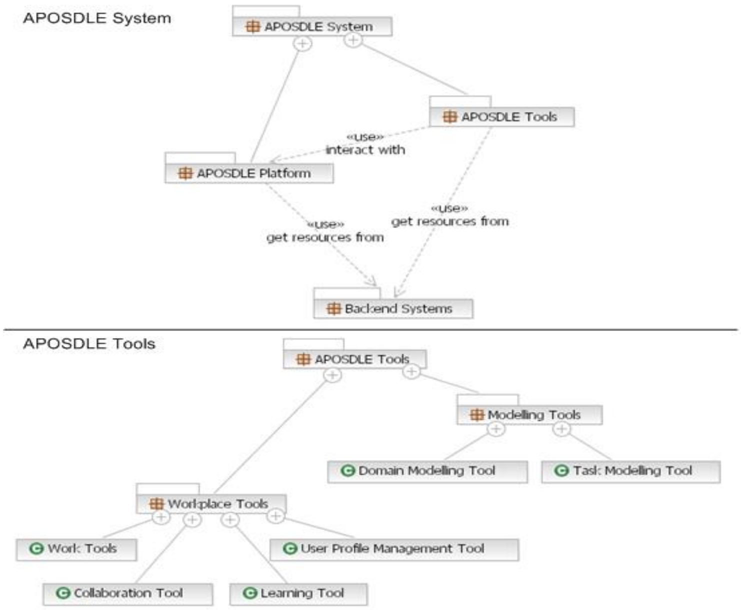

Fig. 1. General Overview on APOSDLE System and on its Tools (Modelling & Workplace) [6]

In short, the current learning context of a knowledge worker is calculated from the current states of those internal models of the system that reflect her current individual task and her advancement in terms of competencies. The result for a system user is then personalised set of recommended learning sources, which, after consumption, might cover an identified context-dependent competency gap. But this result is the outcome of complex calculations in and among the distinct components of the APOSDLE Platform, Tools and Backend Systems. And what if a user wants to understand how all this technical and formal "stuff" has taken place? A scrutable user interface is needed.

2.2 Learning Goals in the Aposdle System

The APOSDLE Sidebar (shown on the left side of figure 2) represents the main user's view on the result of the calculations of the APOSDLE System. For a working task being currently executed by a user, the needed (task-related) competencies are listed by the system. In addition, according to these competencies, a set of learning resources (documents and events) as well as expert contacts are also shown. Thus, the Sidebar expresses a just-in-time personal learning offer for the current working task of a user [7]. The system delivers this most-suitable learning offer based on the current *user context* of the interacting knowledge worker; a user context [9] is described by connecting (at least) competence, domain and task models, as illustrated by the meta-model of user contexts shown on the right side of figure 2.

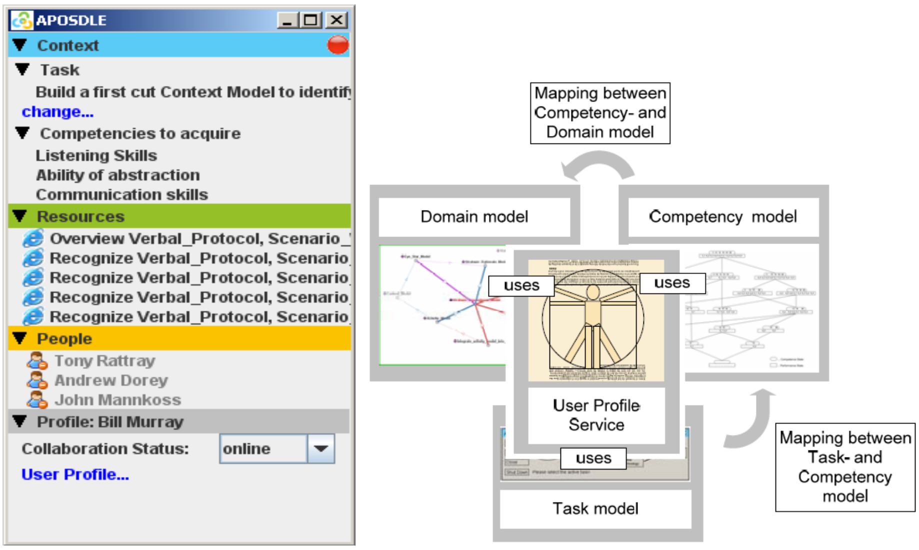

Fig. 2. Left: APOSDLE Sidebar, the User's View on the APOSDLE System. Right: Meta-Model for User Contexts, as used by the User Profile Service. [6]

An individual user context is a dynamic entity continuously derived from the analysis of individual working tasks and individually consumed learning sources. Therefore, *an individual work-place environment reflects distinct individual learning needs at distinct points in time depending on the task at hand*. Within this context, the main goal of the APOSDLE research project is to improve the analysis of learning needs by comparing the tasks already executed by users with those tasks to be faced in the future. For that purpose, user-computer interactions are tracked on real-time to obtain and optimise a fine-grained user model, which in turn builds the computing base for personalised, user-context-based learning recommendations (see as shown in the Sidebar). [6]

From the point of view of the system's usability and user interface design, it is highly important (and challenging) to convey to users a comprehensible explanation of why (and how) these recommendations have been delivered at a certain point in time (past or present). Thus, an intuitive user interface is needed for APOSDLE users in order to scrutinise on the one hand the adaptive behaviour of the system, and on the other hand the history of the tracked observations and inferences of the system in the individual user model. The diagrams on the left side of figure 3 show the relationship of a task model element (*Task*) with a competence model element (*Learning Goal*) as well as the connection of a learning goal with tracked *Learning Activities*, which can be of the type *Learning Event*, *Expert Contacted* or *Document Opened*, depending on the learning resources consumed or experts contacted trough the APOSDLE Sidebar.

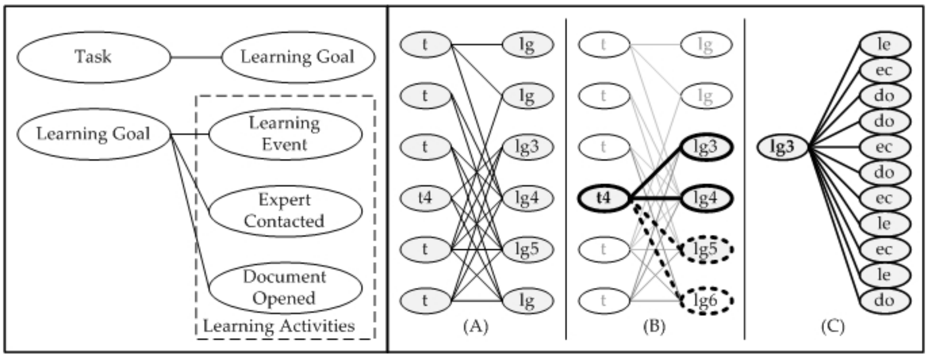

Fig. 3. Left: Relations *Task - Learning Goal* & *Learning Goal - Learning Activities*. Right: An Example of the States of the Models for a Knowledge Worker on the Task "t4".

Given the assumption that the number of model elements behind the user contexts of a medium- to large-size company may be many hundreds, a node diagram in a user interface reflecting an individual user context history might not be clear and comprehensible enough (see "(A)" on right side of figure 3, showing an example of a small part of such a diagram). The diagram "(B)" on figure 3 reflects the adaptive behaviour of the APOSDLE system for a user that performed "Task 4" in the past, whereby the APOSDLE system did not show in its Sidebar all learning goals ("lg3" to "lg6") corresponding to the task ("t4"), rather just those representing her knowledge gap ("lg3" and "lg4"). Furthermore, after tracking the interactions of this user within that task, an additional diagram is needed to show her that she executed some learning activities connected to e.g. "lg3" (see "(C)" in figure 3).

In sum, the learning activities executed in "(C)" mean a competence advancement regarding "lg3" while working on "t4". In turn, due to further connections of "lg3" with other tasks, the system will consider this advancement in the future and eventually suppress the appearance of "lg3" in the Sidebar. In particular this adaptive behaviour of the system should be conveyed to the user in a simple and intuitive way. In APOSDLE, this behaviour is reflected within its Web-based User Profile Management Tool using dynamic lists that show the *working-learning-collaborating history* of individual users.

3. Retracing a Work-integrated Learning Context

APOSDLE's User Profile Management Tool (UPMT) is a Web-based user interface that shows users the contents of their individual user profiles. This chapter focuses on those parts of UPMT presenting to users the states of their competence advancement. The UPMT of the first and second prototypes of the APOSDLE system comprises five sections: *Business Card* (to show personal data, such as name or email address), *Preferences* (including a sub-section for choosing a desired privacy level, and one for selecting preferences about collaboration tools), *Working* (to visualise task-based activities), *Learning* (to visualise issues regarding knowledge acquired), and

Collaborating (to show details on individual collaboration-related events, such as chats, emails or ratings of experts).

The implementation of the APOSDLE system is based on the Java Spring framework [10]. Its Web-based server side is represented by the Tomcat Apache server [11]. To avoid platform dependencies and to enable an AJAX-based client solution [12], the UPMT is built on GWT (Java-based Google Widget Toolkit [13]). The next two sub-chapters introduce main aspects of the UPMT sections *Working* and *Learning* to give users the possibility to retrace their activity history in the context of their competence advancement (based on the models and functionalities shown in the previous chapter).

3.1 Tasks vs. Learning Goals: Knowledge Applied

The section *Working* in UPMT presents APODLE users two visualisation possibilities (*Lists* and *Clouds*, see respectively left and right side of figure 4) to provide a view on the relationships among their performed tasks and the corresponding learning goals. This view is called *Knowledge Applied* and enables users to scrutinise and retrace their competence advancement history as computed and tracked by the system.

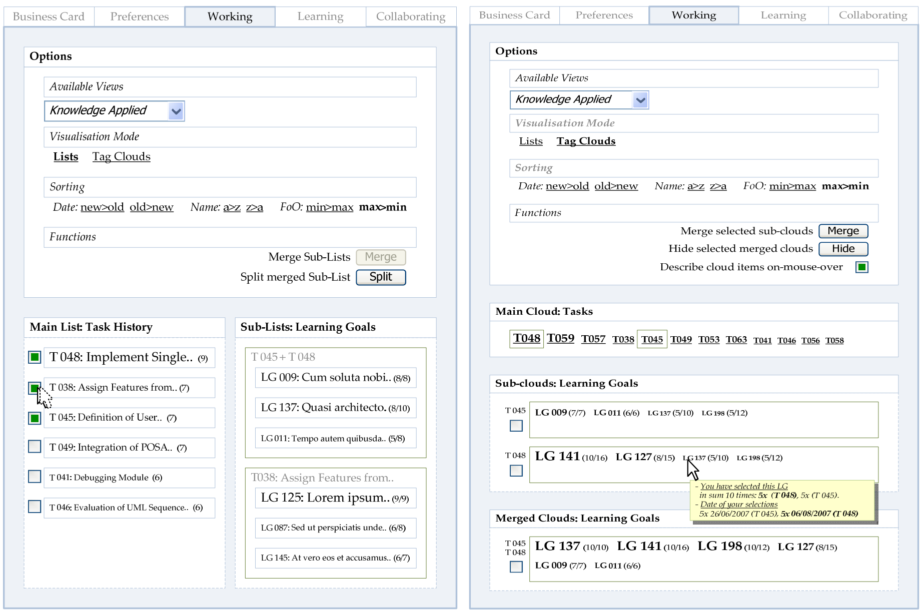

Fig. 4. View “Knowledge Applied” in APOSDLE’s UPMT (left: as *Lists*; right: as *clouds*).

The main area of the Knowledge Applied view shows a user the set of already performed working tasks (“Main List: Task History” in Lists mode or “Main Cloud: Tasks” in Clouds mode). By selecting one task, a second area is dynamically filled with the corresponding learning goals (“Sub-Lists: Learning Goals” or “Sub-clouds: Learning Goals”). The main problem of making such a visualisation scrutable for users relies on the fact that the explanations needed to convey the *APOSDLE-context meaning* of such Task-LearningGoal relationships consist of a large set of descriptive data including temporal, environmental, personal and computational conditions. For example, knowledge workers perform usually the same task several times trough their workflows and thus, depending on their overall task history, they consume continuously distinct learning resources and collaborate with distinct experts in distinct working contexts. This continuous advancement of competencies through their working activities increases at the same time the complexity of the cross-linked relationships among tasks, detected learning goals and recommended learning sources. Furthermore, a consumption of learning events, an opened document or an established collaboration (all of them being tracked by the APOSDLE system) is an indicator for a learning activity, and thus, influences also the degree in which competence advancement is calculated.

For the APOSDLE system, all these factors represent some of the triggers to adapt the elements in the personal Sidebar of a knowledge worker. Thus, the system contains stored numerical values that can be used as key elements to express the *frequency of occurrence* of tasks, learning goals and learning activities. Consequently, these numbers convey intrinsically *individual competence advancement*. This is the reason of utilising the notion of Tag Clouds within the UPMT ([14] [15]). For example, as shown in figure 5, how a learning goal ("LG137") has contributed to competence advancement can be visually retraced through the representative font-size within the clouds.

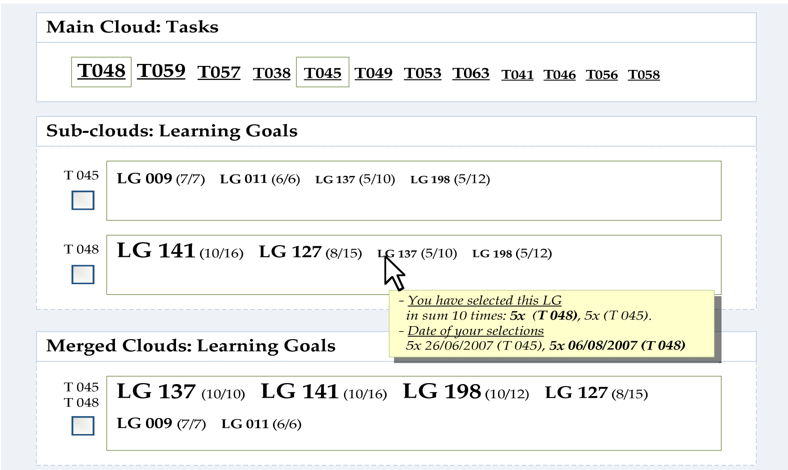

Fig. 5. Segment of the "Clouds" View on "Knowledge Applied" in APOSDLE's UPMT.

On the on hand, "LG137" seems to have had little impact within the task "T 048", but on the other hand, its overall impact on "T 048" and "T 045" is very relevant compared with the other learning goals. Thus, not only from the personal but also from the point of view of the company, filling this gap was essential in the competence advancement of this specific knowledge worker.

3.2 Learning Goals vs. Learning Activities: Knowledge Acquired

The figures in the previous sub-chapter show a way to provide users a possibility of retracing their task history, and simultaneously, a way to scrutinise the relationships of task-related learning goals. APOSDLE's UPMT extends for its users the visibility of personal competence advancement by showing in its section *Learning* the relationships among learning goals and learning activities (see figure 6).

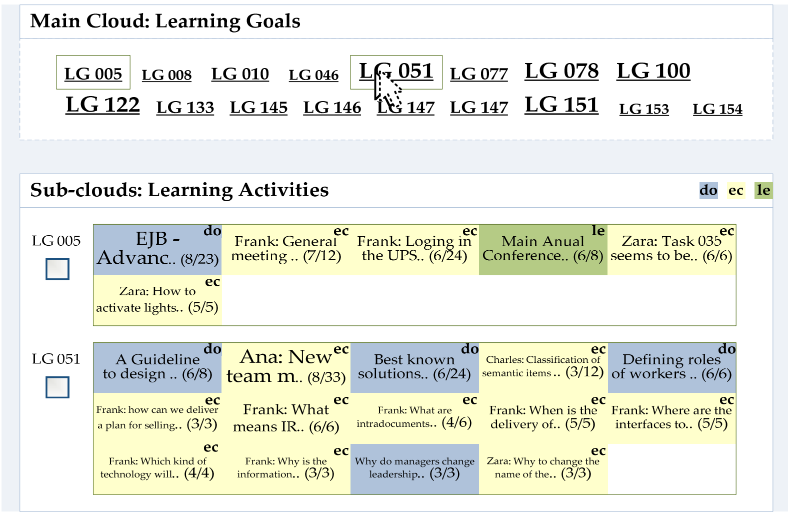

Fig. 6. Segment of the "Clouds" View on "Knowledge Acquired" in APOSDLE's UPMT..

Within APOSDLE, the learning activities of knowledge workers represent distinct ways of acquiring knowledge. On the one hand, the system recommends personal learning sources through the Sidebar depending on the current working task (see figure 2), and on the other side, it tracks which of these recommendations have been accepted and consumed by the users. Thus, the system collects data about consumed learning events, documents opened as well as about the transcripts of collaboration events. As knowledge workers may e.g. read a certain document or chat with a certain expert several times within distinct tasks, these events contribute in a distinct manner to competence advancement.

Through the usage of dynamic font-resizing as in Tag Clouds, the frequency of occurrence of learning activities (i.e. times a learning source has been consumed) conveys to users the impact of their repetitive learning on the knowledge acquired for a certain competence (learning goal). Further, a colour-based visual differentiation of the types of learning activities is also given. For example, the cloud for the learning goal "LG 051" in figure 6 shows that this user has contacted "Ana" (a recommended expert in the knowledge domain of this learning goal) so many times that these collaborations have contributed more to acquiring the needed knowledge than the other learning activities. Further, regarding the learning goals visible in figure 6, this user seems to prefer collaborative events with experts rather than reading documents or consuming learning events.

4. Conclusions and Further Work

The APOSDLE system enables an integrative view on the working environment of knowledge workers through the connection of learning, knowledge and work spaces. This paper has given an overview on how the system deals with the learning context of knowledge workers by means of their competence advancement. Because of the complexity of computing recommendations out of the states of the models in the system, and due to the highly-crossed relationships among model elements, this paper proposes the usage of intuitive cloud-based views on individual states of user models in order to enhance and simplify their visualisation.

APOSDLE'S UPMT views *Knowledge Applied* and *Knowledge Acquired* are provided to users to scrutinise the chronology, relationships and descriptions of their performed tasks, achieved learning goals and executed learning activities. The concept proposed in this paper will be evaluated for the development of the third prototype of the APOSDLE system. Future work will be also the attempt to place learning goals at the centre of a single visualisation with adjacent dynamic relations to tasks and learning activities in order to provide an integrated view on the entire competence advancement of knowledge workers in the APOSDLE context. As the current implementation of the UPMT is a Web-based (in concrete, Ajax-based) solution, it is assumed that (and should be tested if) this cloud-based tool can be reused for integration into other e-learning solutions.

Acknowledgements

The APOSDLE research project (http://www.aposdle.org) is partially funded by the European Community (http://europa.eu.int) under the IST priority (Information Society Technologies, http://cordis.europa.eu/ist) of FP 6 (Sixth Framework Programme for R&D), contract number IST-027023. The support of the following institutions is gratefully acknowledged: Know-Center Graz (http://www.know-center.at), funded by the Austrian Competence Center program Kplus under the auspices of the Austrian Ministry of Transport, Innovation and Technology (http://www.ffg.at) and by the State of Styria, Austria; and Institute for Information Systems and Computer Media (IICM, http://www.iicm.edu), Faculty of Computer Science at Graz University of Technology, Austria.

References

1. ChoiceStream: ChoiceStream Personalization Survey, Consumer Trends and Perceptions, (2007), URL http://www.choicestream.com Last visit: 2008-13-01.
2. Kobsa, A.: Generic User Modeling Systems; in User Modeling and User-Adapted Interaction, Volume 11: Kluwer Academic Publishers, The Netherlands; pp. 49-63, (2001).
3. Hof, R., Green, H., Himmelstein, L.: Now it's YOUR WEB; in Business Week, Issue October 5th 1998; pp. 68-75, (1998).

4. García-Barrios, V.M.: Personalisation in Adaptive E-Learning Systems - A Service-Oriented Solution Approach for Multi-Purpose User Modelling Systems; Dissertation at Institute of Information Systems and Computer Media, Faculty of Computer Science, Graz University of Technology, (2007).
5. Kay, J.: Scrutable adaptation: because we can and must; in Proceedings of 4th International Conference for Adaptive Hypermedia and Adaptive Web-Based Systems, AH 2006, in Dublin, Ireland; Wade, V., Ashman, H., Smyth, B. (eds.), Springer (pub.); pp. 11-19, (2006).
6. Consortium of APOSDLE: First Prototype APOSDLE - Deliverables D1.2, D2.2, D3.2, D4.2, D5.2.; (2007), URL http://www.aposdle.org/results Last visit: 2008-13-01.
7. Bonestroo, W., Ley, T., Kump, B., Lindstaedt, S.: Learn@Work: Competency Advancement with Learning Templates; in Proceedings of LOKMOL-2007, Crete, Greece; pp. 17-20, (2007).
8. Consortium of APOSDLE: APOSDLE Scope and Boundaries - Deliverable D6.1; (2006), URL http://www.aposdle.tugraz.at/media/multimedia/files/aposdle_scope_and_boundaries Last visit 2008-13-01.
9. Ulbrich, A., Scheir, P., Lindstaedt, S.N., Görtz, M.: A Context-Model for Supporting Work-Integrated Learning; in Nejdl, W., Tochtermann, K. (eds.), Innovative Approaches for Learning and Knowledge Sharing, LNCS Vol. 4227, Springer Verlag (pub.), Heidelberg; pp. 525-530, (2006).
10. Spring; Open Source, full-stack Java/JEE application framework; Web-site URL http://www.springframework.org - Last Visit 2007-12-18.
11. Apache Tomcat; Open Source, Java Servlet container; Web-site URL http://tomcat.apache.org - Last Visit 2007-12-15.
12. Garrett J.J.: Ajax: A New Approach to Web Applications; Adaptive Path, LLC, 2005-02-18; Web-site URL http://www.adaptivepath.com/ideas/essays/archives/000385.php - Last visit 2007-12-07.
13. GWT: Google Widget Toolkit; Open Source, Java software development framework for AJAX applications; URL http://code.google.com/webtoolkit - Last Visit 2007-12-05.
14. Hassan-Montero, Y., Herrero-Solana, V.: Improving Tag-Clouds as Visual Information Retrieval Interfaces; In Proceedings of the 1st International Conference on Multidisciplinary Information Sciences and Technologies (InSciT2006), Workshop on Information Visualization; Mérida, Spanien (2006).
15. Rivadeneira, A.W., Gruen, D.M., Muller, M.J., Millen, D.R.: Getting Our Head in the Clouds: Toward Evaluation Studies of Tagclouds; In Proceeding of the 25th International Computer/Human Interaction Conference (CHI2007); San Jose, CA, USA; pp. 995-998, (2007).