Personalized E-Learning through Environment Design and Collaborative Activities

Felix Mödritscher and Fridolin Wild

Institute for Information Systems and New Media,
Vienna University of Economics and Business Administration
Augasse 2-6, 1090 Vienna, Austria
{felix.moedritscher,fridolin.wild}@wu-wien.ac.at

Abstract. Over the last century, many theoretical frameworks and technological solutions for personalized e-learning have emerged. The underlying models, however, are often based on the practice that domain experts develop an adaptation strategy to personalize content or parts of a learning platform, which leads to different problematic aspects decreasing the feasibility or utility of such approaches. After giving a brief overview of the historical development and basic concepts of personalized e-learning, we outline the shortcomings of the traditional 'top-down, ex ante' models and present an alternative approach which deals with personal learning environments, web application mashups, learning activities and learner interactions, as well as pattern-based best practice sharing. Furthermore, a prototypic implementation for our 'learner-driven, bottom-up' approach to personalized e-learning, namely the 'Mash-UP Personal Learning Environment' (MUPPLE), is presented and discussed on the basis of a concrete scenario.

Keywords: Personal Learning Environments, Learning Environment Design, Learner Interaction Scripting, End-User Development.

1 Introduction

According to [1], personalized learning aims at "*tailoring the teaching to individual need, interest and aptitude*" to ensure the most effective knowledge transfer for each learner. In fact, the idea of adapting instructions to learner characteristics has been considered a success factor at least since the 4th century BC, and adaptive tutoring was a wide-spread method of education in the 18th century [2]. Moreover, a first prototypic implementation of an adaptive assessment tool was already reported in 1926, while in the last four decades many technology-based approaches and solutions to personalized learning have emerged [3].

Historically, personalized e-learning is founded on the aptitude-treatment interaction (ATI) research as well as macro and micro-adaptive instructional models. In practice, these streams lead to technologies like Computer-Managed Instruction (CMI), Intelligent Tutoring Systems (ITS), Adaptive Educational Hypermedia (AEH), but they also influenced Learning Management Systems (LMS) and e-learning standards. Referring to the relevant literature [4, 5, 6], personalized adaptive e-learning

A. Holzinger (Ed.): USAB 2008, LNCS 5298, pp. 377–390, 2008.

typically include four types of models: (1) the domain model to describe learning resources and knowledge domain, (2) the pedagogical (learner) model to characterize the learning context and learner states, (3) the didactical model to consider typical teaching aspects, like learning goals, course sequences, didactical requirements, etc., and (4) the adaptation model (rules) to specify the personalization strategies.

Commonly-known frameworks for personalized e-learning address different aspects, such as the conceptual idea (cf. the framework for adaptive e-learning by [4]), the architectural design of an AEH (cf. KnowledgeTree [7]) or a formalization of adaptive behavior (cf. the first-order logic by [8]). Younger trends also focus on standardizing personalization rules, e.g. by using XML-based specifications like IMS Learning Design [9]. However, all these frameworks and their underlying approaches prevent personalized adaptive e-learning from being utilized or utilizable in educational practice. In the next section we discuss the shortcomings of current approaches and examine alternatives based on new concepts and methods from the Internet. Section 3 summarizes our idea of a 'Mash-UP Personal Learning Environment' and a first prototypic implementation. Thereafter, section 4 describes personalization of learning with our prototype on the basis of an exemplary scenario.

2 Shortcomings of Personalized E-Learning and Novel Influences

In learning and research practice, traditional approaches to personalized e-learning still lack important issues as outlined in the following.

First of all, the current definition of personalized (adaptive) e-learning is often restricted to the context of one user interacting with instructions delivered by one system (e.g. the LMS) which, furthermore, contains a pre-defined, up-to-date learner model and automatically adapts to the learner (cf. [3, 4, 6, 10, 11]). In our opinion, this definition is not sufficient to meet the requirements of the real world, and the 'learning environment' includes all possible entities a learner interacts with and all influences on the learning process. More provokingly, we would even say that the learning environment comprises 'everything but the learner'. For the context of e-learning we restrict this definition to all tools on the computer utilized by a learner. Consequently, this point of view widens the scope of adaptivity and personalization. For instance, the knowledge about a learner, a so-called user model, might be distributed over several systems and known by a peer or the facilitator only. Additionally, adaptive behavior is not only observable in one specific system but has to be seen in connection with a learner utilizing various tools to connect to a learning network and collaborate with other actors on shared artifacts. As a result, adaptivity and personalization take place within the whole socio-technical system, the learning network, and not only in one educational system and on the basis of a specific learner model. The adaptation effects, however, are only visible at the frontend of this socio-technical system, precisely at the user interface displaying the learning tools.

Secondly, technology-driven personalized e-learning is based on ex ante, top-down modeling (e.g. modeling of learning styles and cognitive traits depicted in [12]), requiring technological and pedagogical experts to turn a valid adaptation strategy into functions and systemic behavior of an e-learning system, primarily following the paradigm of didactic-awareness [13] and considering aspects of adaptable courseware [14]. However, such strategies to automated adaptation of learning content and functions might lack of validity, e.g. if learners have to self-assess their learning style [15] or if wrong pedagogical assumptions have been made, as shown with a study on interactive media in [16]. Furthermore, new trends in the field of learning research (cf. [17]) try to address the perspective of learners and suggest a learner-driven approach to personalized e-learning, e.g. by analyzing learning behavior and recommending learning experiences of a learning community to learners.

Thirdly, researcher and developer in the field of personalized e-learning often build upon Learning Management Systems (LMSs) or Virtual Learning Environments (VLEs), aiming at adapting parts of such systems, like navigational elements [18] or systemic behavior by orchestrating services [5]. Some approaches even deal with opening up LMSs to externally provide adaptive behavior [19]. Concerning the technological realization, the implementation of personalization in monolithic learning environment requires high efforts in software development and evaluation and often ignores existing tools which are used by learners and might be more efficient in a certain learning context. Younger and more promising approaches focus on the perspectives of learners in terms of personal learning environments composing different learning services into a single user experience [20].

Finally, interoperability is of importance for both e-learning and personalized e-learning, particularly if learners actively contribute to the learning process. If not providing interoperability mechanisms [21] or considering standardized learning content [14, 22], personalisable courseware might get isolated or even be lost within learning platforms. On the other hand, standardizing adaptable courseware extremely increases the complexity and the effort of creating and planning a course, as indicated with a study in [23]. In addition to considering courseware design principles, a heterogeneous landscape of learning tools and services, collaborative learning activities, and learner interaction sequences, facilitators have to deal with learning design and, additionally, with selecting appropriate adaptation models and techniques [18] and extending their courseware, particularly on the basis of existing standards or beyond [14].

All in all, facilitating personalized e-learning experiences can be characterized with technological boundaries, restrictions of existing standards and specifications, as well as with more complexity and efforts for educators. New developments in the Internet, subsumed with the term Web 2.0 [24], aim at overcoming these problematic aspects of e-learning. The effects of the Web 2.0 on technology-enhanced learning have already been examined elsewhere, e.g. in [17, 25]. Following these principles, we propose that, in analogy to Web 2.0 principles, personalized e-learning should be based on '*the Web as a learning space*', allowing learners to use a variety of available tools and content. By providing '*rich learning experiences*' through more interactive user interfaces or community-enabling features, learners can collaborate with peers and actively participate in the learning process, e.g. using blogging or tagging functionality. On the basis of collaborative learning activities and of open, high-quality content (‘*the next Intel inside for learning*’), learners and facilitators can ‘*add value to their learning processes*’, for instance by commenting or tagging learning material or contributing content.

On the other hand, it is also possible to analyze learning behavior in order to ‘*harness the collective intelligence of a networked learning community*’ if personalized e-learning is not restricted to learner interactions with one specific system. Hereby, beneficial semantics for other peers can be provided either by the learners themselves, e.g. by sharing learning experiences with others, or through automated mining techniques trying to extract and exploit the ‘*network effects of a learning community*’, e.g. by recommending tools for learning activities. Finally, from a more technological point of view, these new influences from the Web 2.0 require new development methods for ‘*software above the level of a monolithic LMS*’, being based on ‘*lightweight programming models*’ like RESTful architectures [26], going beyond ‘*the software release cycles for LMSs*’, and considering complex socio-technical processes [27] in order to realize personal learning environments [20]. Thus, the ‘*long tail of software*’ [28] describes the Web 2.0 idea that learners design their own personal learning environments on the basis of available learning tools and services and according to related learning experiences of peers, if given.

Comparing traditional approaches to our idea of ‘**personalized e-learning 2.0**’, we identified significant advantages of this Web 2.0-driven development. Above and beyond, the personalization strategy is shifted from being implemented by a few domain experts and facilitators (ex ante, top-down modeling) to empowering learners to design their personal learning environments, to collaborate with peers, and to transfer learning experiences between facilitators and peers, overall leading to a learning network of actors, artifacts, and activities. Based on learner interactions, this networked community of online learners can be supported by more sophisticated strategies, e.g. for regulating collaborative learning or for reflecting the learning process (learner-driven, bottom-up, just-in-time modeling). Particularly, this would decrease the planning and implementation efforts of personalized e-learning, because modeling is less deterministic and the necessary models which are distributed and partially even outsourced can later be created, through involvement of the learners and on the basis of a valid learning activity design. Moreover, each model can be developed iteratively and, if stable enough, evaluated separately from other models.

While traditional personalized e-learning approaches seem to address a declarative design of procedural experiences, our bottom-up approach aims at designing procedures for creating declarative artifacts, evolving typical learn-by-heart or know-how-to-do goals (low-level learning objectives) to the learn-how-to-learn vision (higher-level learning objectives). Thereby, the idea of the 2.0-driven personalized e-learning also considers that learners can reuse learning experiences by adaptive sharing, cloning, or prototyping learning activities, instead of implementing and partially refining them. Last but not least, we foster the possibility that learners can bring in existing tools and content, instead of working with a given LMS and predefined resources to master teacher-given activities. Concluding this section, we believe the above-mentioned considerations to present a promising approach to personalized e-learning, particularly to enable lifelong learning beyond isolated learning contexts like higher education or workplace learning.

3 The Idea of a Mash-UP Personal Learning Environment

To show personalized e-learning 2.0 in practice, we present the basic concept as well as a first prototypical implementation of a 'Mash-UP Personal Learning Environment' (**MUPPLE**). Hereby, we start off with low-level aspects, such as the technological infrastructure, continue with learner interaction issues, and end up with high-level, learning-related issues behind our idea.

3.1 Technological Infrastructure

As a first step towards a technological infrastructure for a new generation of personalized e-learning solutions, we build upon the Web 2.0 idea of mashups. Hereby, we propose a so-called web application mashup [29] as one possible infrastructure for personal learning environments; learners may also use a portal-like platform with different widgets or different applications on their computer. Extending the idea of traditional mashups, a web application mashup allows displaying various web-based tools into one aggregated view within the browser. Such a solution approach needs to consider the following issues:

- Concluding from mashup visualization techniques [30], the display of different applications next to each other requires a certain (1) *cognitive support for users* (facilitators and peers!) in order to reduce their cognitive load on working with the system. In accordance with iGoogle, Netvibes or other providers of personalized websites, we realized a portal-like OpenACS component, namely the XoMashup application [29], which allows users to arrange tools along a grid layout.
- Addressing (2) *controllability* in the field of personalized e-learning [31], a web application mashup has to give the control over the arrangement of and interaction with the tools to a user. Therefore, our XoMashup component allows a user to rearrange, minimize, maximize, reload and close each window.
- Furthermore, it is possible to launch web applications and even add new ones to the mashup space. As usual, browser-based solutions do cause (3) *technical restrictions*. In our case, it is necessary to start full web applications with all its scripts and style-sheets as a part of the mashup page. Thus, we implemented our mashup solution on the basis of 'iframes'. This may be the only way to guarantee an own environment for each tool but may not be supported by all browsers. Further, the usage of iframes enforces the prevention of DOM operations which would reset the content of an iframe. Consequently, the grid-based windowing system of XoMashup is realized with absolute positioning and the manipulation of CSS directives.

All in all, the web application mashup solution allows learners to reuse existing (web-based) tools and services and can be considered as a technological infrastructure for our approach. Moreover, web application mashups are very flexible and, therefore, useful for many other application areas as well. Nevertheless, without some kind of underlying semantics like necessary models for personalizing the learning process, the XoMashup component would be nothing more than a personalizable (customizable) portal system, lacking pedagogical support for learners such as guidance or reflection.

3.2 Learner Interaction Model

Therefore, we built up a learner interaction model for describing how learners can design their personal learning environment and interact with it. The following aspects were taken into consideration for this model:

- In order to be (4) *independent of a subject domain*, we applied the Activity Theory model in a similar way as manifested for the INCENSE system [32]. Basically, we broke down the learning context into situations which describe the physical and social environment of learners. In such a situation, a learner is engaged in a so-called activity which consists of actions and objects (artifacts or other outcomes) and includes tools (or tool combinations) and other actors (facilitators or peers, even in multiple roles). Such a learning activity is meant to be our basic instructional entity where learners actively experience a domain and construct knowledge.
- This notion of a learning activity is (5) *simple and understandable for learners*, thus is considered to be of importance for a (6) *scrutable systemic behavior* [33] and a good basis for experiencing further personalization strategies.
- To enable (7) *reflective learning* [34](p.7), we decided to bind each action to one specific tool and one specific object in order to produce one outcome. Although different actors can work on the same action and even produce the same outcome, each learner only sees her own actions, and all started actions are visualized together with the corresponding object and tool.
- Additionally, these action-object-tool triples are recommended to peers on defining and starting new actions as a certain (8) *learner support*. However, learners are able to overwrite decisions and recommendations given by the system and may build up their personal learning environment by defining own actions and objects, bringing in own tools, and going through the actions in their own sequence to achieve the outcomes.
- Addressing (9) *learnability and efficiency*, our learner interaction model is implemented in the form of a domain-specific language called 'Learner Interaction Scripting Language' (LISL). Table 1 shows an example of a simple activity consisting of two actions. First, the learner is expected to record a short self-description with the tool VideoWiki (http://distance.ktu.lt/videowiki/), whereby the REST-based call for this action has to be specified and the URL for the object 'self-description' is determined by completing the action. Second, the learner should go through the self-descriptions of the peers by accessing a predefined URL, e.g. the collection containing all self-descriptions.

Table 1. Exemplary LISL code for the activity ‘Getting to know each other’ consisting of two action statements, each one bound to one object and one tool

- ➢ define action Compose with url http://[...]
- ➢ define action Browse
- ➢ define object ‘self-description’
- ➢ define object ‘descriptions of peers’ with url […]
- ➢ define tool VideoWiki with url http://[...]
- ➢ Compose ‘self-description’ using VideoWiki
- ➢ Browse ‘descriptions of peers’ using VideoWiki

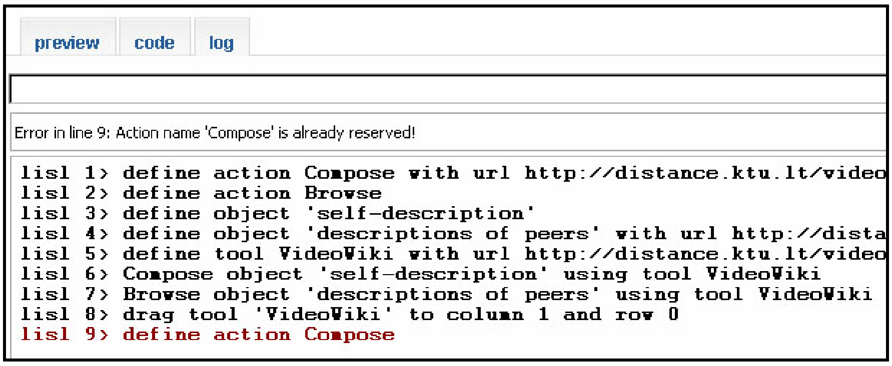

Fig. 1. Web-based LISL interpreter displaying the interpreted LISL script from Table 1 (red lines indicate errors, whereby the error message is shown below the command input field)

We build a web-based interpreter for LISL code, as can be seen in Fig. 1. This scripting approach allows experienced users to code their learning activities very efficiently, while novices can use web-based control widgets and dialogs which act as a wrapper for the LISL statements and are shown in the upcoming section.

3.3 Higher-Level Learning Paradigms

On top of the mashup infrastructure and the learner interaction model, higher-level learning paradigms address important issues for building and sustaining networked communities of learners:

– With regard to [35], the constructivistic-collaborative approach to adaptive e-learning deals with aspects of the (10) *active participation of learners*, motivational factors like self-esteem and (11) *collaborative activities*. Technologically, such considerations lead to a high degree of interactivity also found in games and simulations, to adaptation strategies for motivating learners, e.g. by pedagogical agents, or to adaptivity through collaboration, most prominently addressed by the field of Computer-Supported Collaborative Learning (CSCL). As already mentioned, the MUPPLE approach stresses a method to build and sustain a learning network of actors, artifacts, and activities, which increases the motivation to learn and aims at developing more complex competencies [36]. Similar to Web

2.0-driven platforms like Facebook, learners require facilities to get involved into collaborative activities and to regulate collaboration and social interactions with peers.

- Particularly for attracting new users, the success of MUPPLE highly depends on recognizable benefits for learners. Hereby, we foster the paradigm of (12) *best practice sharing* on the basis of activity patterns which can be provided by facilitators and learners to be shared within the community. Similarly to the idea of scripting collaborative activities [37], we apply the LISL scripting language to describe these activity patterns. As learning activities are encoded in form of LISL scripts, they can be exported into activity patterns and shared with other learners; vice versa, peers can use available best practices and create their own activities out of these patterns.
- Beside learner-driven best practice sharing, the bottom-up approach of MUPPLE supports the (13) *analysis of former learning scripts* in order to personalize different aspects of learning, e.g. by recommending action-object-tool triples to inexperienced learners or suggesting tool landscapes for certain activities. Such personalization strategies also address the learnability and efficiency of applying MUPPLE in practice.

4 Personalized Learning with MUPPLE

Based on our wider understanding of personalized e-learning, we build a first prototype considering the issues mentioned so far. Fig. 2 shows a screenshot of the learner's view on our exemplary activity 'Getting to know each other' which consists of the two actions 'Compose self-description' and 'Browse descriptions of peers' (cf. table 1). On the top, the header displays the activity currently opened. To the left-hand side, learners are supplied with an overview of their own activities and can navigate through them. By clicking on it, a learning activity is loaded and displayed in the content area; additionally, a branch with all action-object-tool triples included is opened simultaneously. The content area provides three different view modes of a MUPPLE page. By choosing one of the three tabs, a learner has a view on the web application mashup ('preview'), an editor for the LISL code of this page ('code') or the LISL interpreter ('log'). This structure of a MUPPLE page is related to principles of end-user development, which is closer examined in [38].

The LISL interpreter does not only show the interpreted code, but also highlights possible errors with detailed explanations and allows entering single lines of code (cf. Fig. 1). The preview mode, on the other hand, comprises an integrated view of all learning tools launched so far. Each tool is located within an own window (a so-called 'Mupplet') with the control elements mentioned in the last section ('reload original URL', 'minimize', 'maximize', and 'close') on the upper right side. Creation facilities on the left-hand side allow creating new activity pages from blank or from given patterns. Furthermore, it is possible to add action-object-tool triples to an opened MUPPLE page, whereas possible values are recommended on the basis of all other

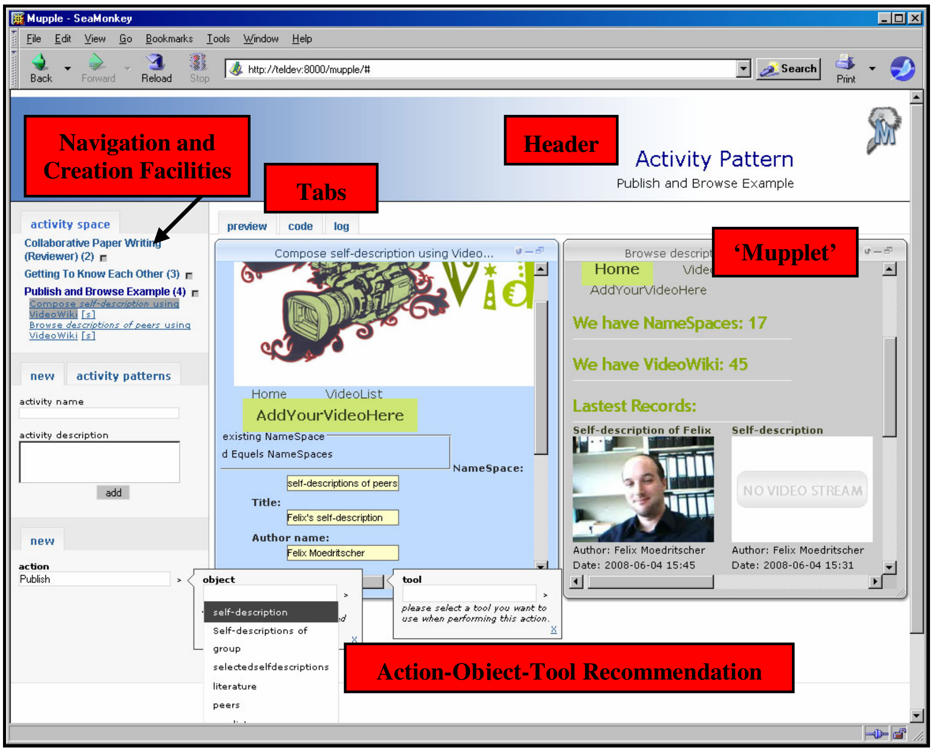

Fig. 2. User view on activity 'Getting to know each other', where the screen consists of a header (top), facilities for navigation and activity management (left), and the content area with its three tab-views (the mashup space, the LISL code editor, and the command-line interpreter).

activity pages. A ranking strategy has not been implemented for these recommendations so far. However, features like the closeness between activities (e.g. through derivation from the same pattern), the action-object-tool binding or social relations in the learner network may present useful factors to rank and filter values if there are too many of them.

In the following, a simple scenario is described to explain how personalization of the learning process takes place on working with the MUPPLE prototype: Consider a group of students distributed over various universities in different countries. To introduce the students to each other, the facilitator decides to use MUPPLE and predefines an activity pattern for this purpose. Based on former experiences with such a socialization exercise, this pattern includes the following action statements:

– Compose 'self-description' using VideoWiki
– Browse 'descriptions of peers' using VideoWiki
– Contact 'two peers' using WebMail
– Delete 'spam entries' using VideoWiki

After introducing MUPPLE to the students, the facilitator invites them to create a MUPPLE page from the given pattern. Now, each student instantiates a learning activity 'Getting to know each other' from the pattern and, furthermore, customizes it according to her own need. For instance, if students do not need the 'delete' action and might not even have appropriate permissions within the VideoWiki tool, they simply can remove this action from their pages. A possible result could be the MUPPLE page shown in Fig. 2. On the other hand, the facilitator who obviously has a different role in this collaborative activity uses the action 'delete' to clean up the collection containing the self-descriptions from spam regularly. With respect to controllability, a student might want to use a web-based chat tool to communicate with peers. So, she could bring in this tool and remove the original one, either by modifying the LISL code or using web-based control widgets. Finally, personalization takes place on creating new action statements if MUPPLE recommends action-object-tool triples from peers. Consequently, MUPPLE also might provide observation facilities to inform users about changes in the activities of peers or the facilitator, which are currently not realized.

Beside personalization within the learning activities, MUPPLE and its pattern-based best practice sharing method allow personalized learning in a broader sense. First, improved patterns can be derived from the learning activities experienced in practice, which, in our opinion, might increase the quality of the content and activities within MUPPLE. Second, the activity patterns are valuable for other learners and peer groups as well. For instance, other facilitators could reuse the learning experiences of the above-mentioned scenario if the actors share them in terms of an activity pattern. Hereby, the pattern-based approach might be useful to avoid the cold-start problem of learning platforms. Third, this best practice sharing method is also advantageous with respect to scrutability and privacy, two important issues for personalization. Activity patterns and their underlying semantic model are simple to understand, and exporting a LISL script easily allows filtering learner-specific lines of code. Here, the learners can regulate the amount of a (successful) activity they want to share. Above and beyond, personalization and (automated) adaptation effects take place within the learning environment (i.e. the learning network) while learners (peers and facilitators) use tools to collaborate on shared artifacts within their common activities.

Despite of the possibilities and strengths of our MUPPLE approach, a few disadvantages have to be outlined here. Primarily, these problems concern technological issues. First of all, it is necessary to have a high degree of interoperability between web applications, which now is not always the case. This specifically relates to single-sign-on procedures and communication channels to transfer both data and events from one application to the other. For example, the WebMail client requires authentication, so, currently, learners have to login separately in each application. Regarding communication channels, [21] propose a specification how to realize distributed feed networks with buffered-push capabilities. We intend to further investigate these means and will gain experiences on how they can be incorporated into LISL. It is planned to introduce additional 'connect' statements for combining tools with the abovementioned feed-based interoperability mechanism. We can think of other approaches, though, and we do not have a solution for the efficient communication of events. Secondly, the utilization of iframes causes problems in cross-domain scripting (cf. [39]). Finally, we are also aware that LISL and MUPPLE still lack important functions, especially in the area of regulating collaboration and privacy, and a comprehensive evaluation.

5 Conclusions and Future Work

In this paper, we stated that personalization and adaptivity is much more complex in real world learning situations, particularly if learners connect to a network of actors, artifacts, and activities. Therefore, we consider traditional approaches for personalized e-learning as not sufficiently to provide personalized learning experiences on the computer. As a consequence, we introduced our idea of 'personalized e-learning 2.0', taking into consideration that learners connect to a socio-technical network and collaborate on shared artifacts and outcomes. Although our prototypical implementation of a Mash-UP Personal Learning Environments (MUPPLE) is work in progress, we described how the learning process is adapted through environment design and collaborative activities in networked communities. As a conclusion, we underline the following three success factors being comprised in our model:

– First, we break up with traditional personalized e-learning models and consider the learning environment not to a pre-condition for, but an outcome of personalized e-learning. Therefore, we build upon learning environment design and our learner interaction scripting language to be able to describe, understand, and reproduce the outcome of learning.
– Second, the pedagogical model behind MUPPLE is very simple, so that learners can understand how personalization works. Additionally, this activity model is a solid basis for learning environment design and further personalization strategies, like the automated analysis of user behavior and network effects as well as the provision of recommendations or advanced regulation facilities.
– Third, the structure and domain-independency of these learning activities address higher-level learning objectives, independently of a subject domain, and enable best practice sharing as well as reflective learning, i.e. aiming at paradigms like learn-how-to-learn instead of learn-by-heart.

In total, we believe that personalized e-learning will proceed from an instructional design and top-down, ex ante modeling to a learner-driven, bottom-up, just-in-time adaptation of learning by considering the principles of Web 2.0. This is meant to be 'personalized e-learning 2.0'. Our future work will address interoperability issues of learning tools, regulation facilities for collaborative activities in learning networks as well as experiences in real-world learning settings, particularly to evaluate the utility of the MUPPLE approach for higher education and lifelong learning. Concerning personalized e-learning, we have to think of further strategies to recommend (advertise) pattern-based best practices to peers and to support learners in their collaborative activities within their learning network.

Acknowledgements

This work has been produced in the context of iCamp, a research and development project financially supported by the European Union under the ICT programme of the 6th Framework Programme (Contract number: 027163).

References

1. Heller, J., Steiner, C., Hockemeyer, C., Albert, D.: Competence-Based Knowledge Structures for Personalised Learning. International Journal on E-Learning 5(1), 75–88 (2006)
2. Sampson, D., Karagiannidis, C.: Kinshuk: Personalised Learning: Educational, Technological and Standardisation Perspective. Interactive Educational Multimedia 4, 24–39 (2002)
3. Park, O., Lee., J.: Adaptive Instructional Systems. In: Jonassen, D.H. (ed.) Handbook of Research on Educational Communications and Technology, pp. 651–684. Lawrence Erlbaum Associates, Mahwah (2004)
4. Shute, V., Towle, B.: Adaptive E-Learning. Educational Psychologist 38(2), 105–114 (2003)
5. Conlan, O.: The Multi-Model, Metadata Driven Approach to Personalised eLearning Services. University of Dublin, Dublin (2005)
6. Mödritscher, F.: Adaptive E-Learning Environments: Theory, Practice, and Experience. VDM, Saarbrücken (2008)
7. Brusilovsky, P.: KnowledgeTree: A Distributed Architecture for Adaptive E-Learning. In: Proceedings of the World Wide Web Conference (WWW), pp. 104–113 (2004)
8. Henze, N., Nejdl, W.: A Logical Characterization of Adaptive Educational Hypermedia. New Review of Hypermedia and Multimedia 10, 77–113 (2004)
9. Specht, M., Burgos, D.: Modeling Adaptive Educational Methods with IMS Learning Design. Journal of Interactive Media in Education (Adaptation and IMS Learning Design), 1–13 (2007)
10. Henze, N., Dolog, P., Nejdl, W.: Reasoning and Ontologies for Personalized E-Learning in the Semantic Web. Educational Technology & Society 7(4), 82–97 (2004)
11. Chen, C.-M., Lee, H.-M., Chen, Y.-H.: Personalized e-learning system using Item Response Theory. Computers & Education 44(3), 237–255 (2005)
12. Graf, S., Lin, T., Kinshuk, J.: The relationship between learning styles and cognitive traits - Getting additional information for improving student modelling. Computers in Human Behavior 24(2), 122–137 (2008)
13. Helic, D.: A Didactics-Aware Approach to Management of Learning Scenarios in ELearning Systems. Graz University of Technology, Graz (2006)
14. Mödritscher, F., García-Barrios, V.M., Gütl, C.: Enhancement of SCORM to support adaptive E-Learning within the Scope of the Research Project AdeLE. In: Proceedings of the World Conference on E-Learning in Corporate, Government, Healthcare, & Higher Education (E-Learn), pp. 2499–2505 (2004)
15. Phillips, M.: Cognitive Style's influence on Media Preference: Does it matter or do they know? In: Proceedings of the World Conference on Educational Multimedia, Hypermedia & Telecommunications (ED-MEDIA), pp. 1023–1028 (2005)
16. Holzinger, A., Kickmeier-Rust, M., Albert, D.: Dynamic Media in Computer Science Education; Content Complexity and Learning Performance: Is Less More? Educational Technology & Society 11(1), 279–290 (2008)

17. Rollett, H., Lux, M., Strohmaier, M., Dösinger, G., Tochtermann, K.: The Web 2.0 way of learning with technologies. International Journal of Learning Technology 3(1), 87–107 (2007)
18. Brusilovsky, P.: Methods and techniques of adaptive hypermedia. User Modeling and User-Adapted Interaction (UMUAI) 6(2-3), 87–129 (1996)
19. Gütl, C., Mödritscher, F.: Towards a Generic Adaptive System applicable for Web-based Learning Management Environments. In: Proceedings of the Annual Workshop of the SIG Adaptivity and User Modeling in Interactive Systems (ABIS), pp. 26–31 (2005)
20. Van Harmelen, M.: Personal Learning Environments. In: Proceedings of the International Conference on Advanced Learning Technologies (ICALT), pp. 815–816 (2006)
21. Wild, F., Sigurdarson, S.E.: Distributed Feed Networks for Learning. European Journal for the Informatics Professional (UPGRADE) 9(3) (2008)
22. Santos, O.C., Boticario, J.G.: Meaningful pedagogy via covering the entire life cycle of adaptive eLearning in terms of a pervasive use of educational standards: The aLFanet experience. In: Tomadaki, E., Scott, P. (eds.) EC-TEL 2006. LNCS, vol. 4227, pp. 691–696. Springer, Heidelberg (2006)
23. Mödritscher, F.: e-Learning Theories in Practice: A Comparison of three Methods. Journal of Universal Science and Technology of Learning (JUSTL) 0(0), 3–18 (2006)
24. O'Reilly, T.: What is Web 2.0. O'Reilly Media Inc. (2005) (2008-07-21), http://www.oreilly.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html
25. Ebner, M., Holzinger, A., Maurer, H.: Web 2.0 Technology: Future Interfaces for Technology Enhanced Learning? In: Stephanidis, C. (ed.) HCI 2007. LNCS, vol. 4556, pp. 559–568. Springer, Heidelberg (2007)
26. Lathem, J., Gomadam, K., Sheth, A.: SA-REST and (S)mashups: Adding Semantics to RESTful Services. In: Proceedings of the International Conference on Semantic Computing (ICSC 2007), pp. 469–476 (2007)
27. Jones, S., Maiden, N.A.M.: RESCUE: An Integrated Method for Specifying Requirements for Complex Socio-Technical Systems. In: Mate, J.L., Silva, A. (eds.) Requirements Engineering for Socio-Technical Systems, pp. 245–265. Idea Publishing (2004)
28. Kraus, J.: The long tail of software: Millions of Markets of Dozens. Bnoopy: An entrepreneurship blog (2005) (2008-07-21), http://bnoopy.typepad.com/bnoopy/2005/03/the_long_tail_o.html
29. Mödritscher, F., Neumann, G., García-Barrios, V.M., Wild, F.: A Web Application Mashup Approach for eLearning. In: Proceedings of the OpenACS and.LRN Conference, pp. 105–110 (2008)
30. Spoerri, A.: Visual Mashup of Text and Media Search Results. In: Proceedings of the International Conference Information Visualization, pp. 216–222 (2007)
31. García-Barrios, V.M., Mödritscher, F., Gütl, C.: Personalisation versus Adaptation? A User-centred Model Approach and its Application. In: Proceedings of the International Conference on Knowledge Management (I-KNOW), pp. 120–127 (2005)
32. Akhras, F.N., Self, J.A.: System Intelligence in Constructivist Learning. International Journal of Artificial Intelligence in Education 11(4), 344–376 (2000)
33. Kay, J.: Stereotypes, Student Models and Scrutability. In: Proceedings of the International Conference on Intelligent Tutoring Systems (ITS), pp. 19–30 (2000)
34. Boud, D., Keogh, R., Walker, D. (eds.): Reflection: Turning Experience Into Learning. Routledge, Abingdon (1985)
35. Koper, R., Olivier, B.: Representing the Learning Design of Units of Learning. Educational Technology & Society 7(3), 97–111 (2004)

36. Mödritscher, F., García-Barrios, V.M., Gütl, C.: The Past, the Present and the Future of adaptive E-Learning: An Approach within the Scope of the Research Project AdeLE. In: Proceedings of the International Conference on Interactive Computer Aided Learning, ICL (2004)
37. Dillenbourg, P., Jermann, P.: Designing integrative scripts. In: Fischer, F., Mandl, H., Haake, J., Kollar, I. (eds.) Scripting computer-supported collaborative learning: Cognitive, computational and educational perspectives, pp. 277–302. Springer, New York (2007)
38. Wild, F., Mödritscher, F., Sigurdarson, S.E.: Designing for Change: Mash-Up Personal Learning Environments. eLearning Papers 9 (2008)
39. Jackson, C., Wang, H.J.: Subspace: Secure Cross-Domain Communication for Web Mashups. In: Proceedings of the International Conference on World Wide Web (WWW), pp. 611–620 (2007)