Toward Developing a Semantic Mash-up Personal and Pervasive Learning Environment: SMupple

Ahmet Soylu[1], Fridolin Wild[2], Felix Mödritscher[3], Patrick De Causmaecker[1]

[1] K. U. Leuven, Department of Computer Science, CODeS, iTec, Kortrijk, Belgium, {Ahmet.Soylu, Patrick.DeCausmaecker}@kuleuven-kortrijk.be
[2] The Open University, Knowledge Media Institute, Milton Keynes, United Kingdom, f.wild@open.ac.uk
[3] Vienna University of Economics and Business, Department of Information Systems, Vienna, Austria
felix.moedritscher@wu.ac.at

Abstract. Personal Learning Environments have emerged as a complementary, even challenging, paradigm to Adaptive Learning Systems. One can argue that Pervasive Learning Environments aim at replacing the physical learning environment while Adaptive Learning Systems focus on replacing the instructor. We believe that amalgamation of these two approaches in a complementary manner, i.e. through setting an appropriate balance between learner and system control, is promising. Consequently, we consider mash-ups to be crucial for a successful realization of digital personal learning environments. However, mash-ups are also accompanied by critical technical and usability challenges. In this paper, we try to identify some of these challenges, present solution approaches from a conceptual point of view, and describe our Semantic Mash-up Personal and Pervasive Learning Environment (SMupple) proposal along with initial implementation and evaluation details.

Keywords: Personal Learning Environments, Mash-ups, Ontologies, Embedded Semantics, Workflows, Pervasive Computing.

1 Introduction

In general, Adaptive Learning Systems (ALSs) focus on automatically, often intrusively, changing the system behavior according to the learner's needs and other characteristics, aiming at adapting the learning material and its presentation. However, it is apparent that it is not possible to predefine adaptation rules for all different usage contexts. Furthermore, Wild and his colleagues [1] claim that adaptation technologies take away experiences from end-users (learners) thus prohibiting the development of important competences. In this respect, Personal Learning Environments (PLEs) emerge as a complementary, even challenging, paradigm to the ALSs. Wild et al [1] value learning environment as an important aspect of the learning process and consider it as an output of learning rather than a mere input. Digital learning environments can be composed of different applications, artifacts, and actors. The individual at the centre modifies this environment through interacting with it, intending to positively influence her social, self, methodological, and professional competences and to change her potentials for future action. In other words, a learner actively or passively creates her own personal learning environment. In short, one can argue that PLEs aim at replacing the physical learning environment while ALSs focus on replacing the instructor.

Considering PLEs, learners acknowledge the abundance and variety of web applications, services and data sources to be used within their environments. Moreover, different technological devices, like mobile phones, digital media solutions, tablet PCs, intelligent household appliances, etc. are expected to be connected to the Web and serve their functionalities through embedded web servers or gateways coupled with the internal functions of available devices, possibly, through RESTful APIs [2]. This leads us to extend PLE paradigm to Personal and Pervasive Learning. Here, mash-up approaches enable users to design their ubiquitous and personal learning environments through combining functionalities and data available on the Web. However this brings in some challenges. In this paper, we identify these challenges and present our solution approach, which builds on semantic technologies and, referring to [1], is called Semantic Mash-up Personal and Pervasive Learning Environments (SMupple).

The rest of the paper is structured as follows. In section 2, we provide a discussion on mash-up approach with respect to ALSs. In section 3, we elaborate the challenges identified and present our approach and proposal. In section 4, we outline an implementation and evaluation plan before the approach is discussed on the basis of related work.

2 Mash-ups and Adaptive Learning

ALSs [3] have been an active research area for several decades, trying to offer user-tailored learning experiences based on various adaptation techniques often realized through different Artificial Intelligence (AI) and Machine Learning (ML) approaches (e.g., Intelligent Tutoring Systems – ITSs). One can claim that ALSs aim at replacing or replicating human instructor by a machine, in many cases with superior competences due to their obvious data processing and computational power. Although we acknowledge the appropriateness of such an approach to some extent, it is still arguable on a theoretical and pragmatic level.

On the one hand, Sharples et al. [4] argue that an "intelligent" system cannot substitute a teacher or a facilitator; it can only keep limited dialogue at the level of actions, and it has no capabilities to explore student's misunderstandings or to help them to reach a shared understanding. This implies that, in a digital learning environment, learners should get a chance to develop important skills towards exploring and managing their learning processes, possibly also with the help of peers and facilitators available in their digital social networks. On the other hand, a user-to-system view of adaptation (e.g., intelligent tutoring) reflects a producer-consumer model of learning (i.e., classroom model) where teachers act as content producers and students act as content consumers. In other words, an adaptive system considers students as proprietary end-point machines which will perform smoothly if the producer machine feeds them with appropriate content in an appropriate way. Therefore, it puts an overloaded emphasis on content and presentation adaptation while ignoring the self-organizational skills of the learners and the learning environment.

From a pragmatic point of view, adaptation effects in adaptive systems can be considered as a mapping between profile/model space of the learner (i.e., the context space in a broader sense) and the adaptive behavior space. However, it is not always feasible to predefine these mappings for all possible learner characteristic and adaptation pairs. Therefore, it becomes crucial to enable learners to shape and control their environments to some extent. These critiques become more apparent with the current shifts in computing itself. For instance, with the emergence of Pervasive Computing (PerCom) [5], researchers have been trying to realize computing systems and applications which can seamlessly immerse into the users' daily life. Similar to adaptive systems, PerCom aims at adapting an environment according to specific usage contexts rather than only considering user/learner characteristics. In this respect, we argue that dominant machine control can cover and is appropriate only for a limited amount of cases. Therefore it is required to put users/learners into centre stage and to provide them with "intelligent" guidance, support, and awareness through non-invasive adaptation mechanisms.

Indeed, PerCom does a great job by moving attention of researchers to the notion of the environment. The interaction model of learners with their environments shall not be considered as a one-dimensional one; it is multi-dimensional due to the composite nature of their real learning environments which comprise other actors, artifacts, activities, and communities. Even in the case of being provided with a single "intelligent" application, one should accept that learners keep interacting with peers, instructors, friends etc. and that they search and consume other relevant content outside their main learning platform through various other applications (i.e., Google, Facebook, Doodle etc.). Upon that fact, the empowerment of learners to shape their environments by orchestrating the applications and data sources available is promising. The mash-up paradigm has emerged as a key solution proposal to this demand but is also accompanied with new challenges. Mash-ups are complementary to ALSs since they continuously involve learners/users and shift user control to learners instead of providing strong invasive adaptations.

3 Mash-Up based Approach towards Personal and Pervasive Learning Environments

We consider the mash-up paradigm to be crucial for realizing the PLE vision within the infinite space of the Web. In this context, we believe that a conceptual description of a personal learning environment and the identification of basic requirements for a digital PLE shall be useful for situating important challenges. On a conceptual level, a learning environment can be seen as space of entities, including people, artifacts, tools, learning objects etc. available to the learner. Each of these entities is attached with several possible activities. Additionally, composite activities and composite entities encompass several other entity-activity pairs and entities respectively. In that space learners derive their personal (sub-) environments, orchestrate member entities for their goals through maintaining data and interaction flows between these entities, and continuously refine the PLEs as a result of their activities and often through their own implicit formative assessment methods.

A PLE can be further partitioned into disjoint or overlapping clusters with respect to varying goals of learners. Learners often shift their focus from one cluster to another according to their current goals. From this perspective, a mash-up personal and pervasive learning environment enables learners to construct their digital learning environments spanning various digital web resources and web-enabled devices encapsulated through widget like constructs.

Considering mash-ups, they can be created at the client-side (i.e., in a browser) or at the server-side. We identify two different types of mash-ups: (1) dashboard type (e.g., [1]), (2) box type (e.g., [6]). The former is usually created at the client-side where different applications are shown in the learner's browser as widgets (all visible). Data and events can be transferred from one widget to another one mainly through inter-widget communication on the client, occasionally also through server-sided synchronization mechanisms. The latter mash-up type is usually created and provided by a server and combines the different applications into one single user experience (only the resulting application is visible). Data and events can be distributed among the applications through server-sided synchronization mechanisms. The end product can also be used for developing mash-up hybrids of both types.

With respect to above descriptions and by considering the existing implementations [7], we identified several challenges. These challenges and our approach is described in three tiers which is partially depicted in Fig. 1.

Seven particular challenges have been identified each mapping to at least one tier: (1) composition/integration (services, applications and data), (2) inter widget communication. The first two challenges deal with data links between different applications, through server-sided synchronization or inter widget communication based on syntactic means, which is not sufficient for automated integration and composition of services and leaves a huge burden to the end-user. Accordingly, injecting semantics through ontologies and embedded semantics technologies (i.e., microformats, RDFa, eRDF) may serve well for automated linking (e.g., [6, 8]). (3) Workflow management: this challenge is related to typical mash-up composition and requires users to define full workflows thus cognitively overloading them. Similarly to linking data manually, an approach which requires modeling of complete workflows is unrealistic because hardly any end-user is capable in creating full workflow models for everyday tasks. However a machine observer can extract simple workflows or at least fragments of them. Therefore, we foster the idea of enabling mash-up PLEs on the basis of incomplete workflows automatically generated from user interaction recordings captured. (4) Adaptive guidance and support: the fourth challenge is necessitated from the fact that involving users in the design and development of their environments requires adequate machine support, in terms of non-invasive adaptations and recommendations. At this point, a formalized representation of the user's context through an ontology is promising with respect to "intelligent" guidance and end-user development [9]. (5) Environment awareness and control [10]: in physical environments users manage a limited number of entities with a relatively high awareness, however the Web offers an almost infinite amount of resources; therefore it is crucial to maintain awareness and control of one's space, so that the links between a learner and the environment stay tight. (6) Ease of orchestration: since the learner is confronted with more resources, learners should not experience a cognitive overload while managing the space.

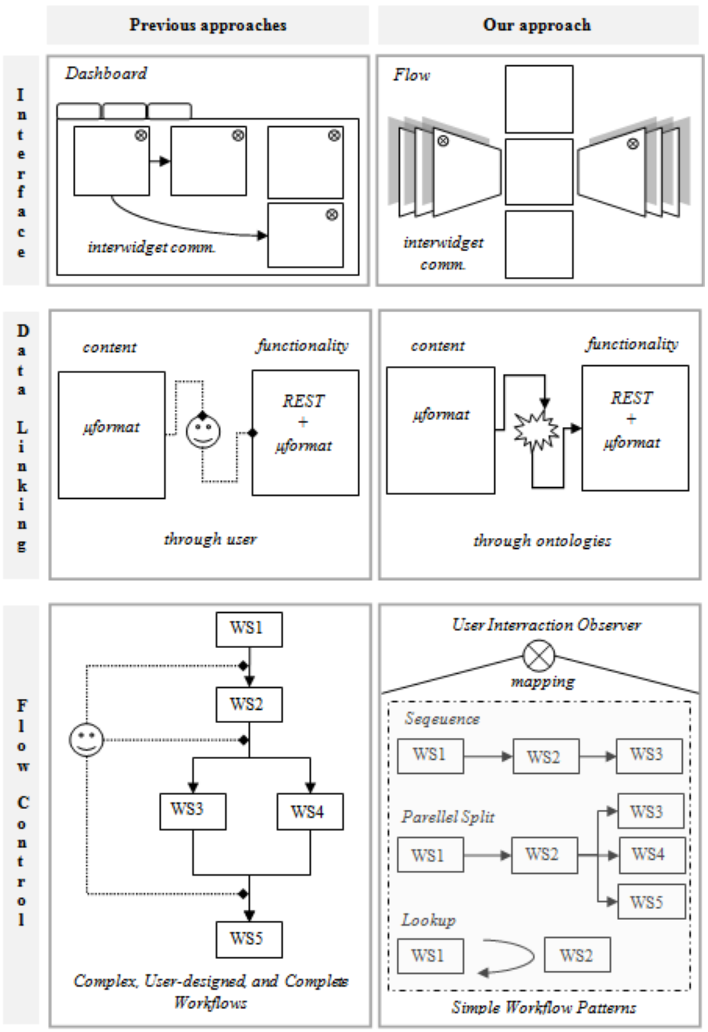

Figure 1. Presentation and comparison of different PLE approaches along the three tiers.

(7) Engaging learner experience: learners should feel comfortable through their experiences with PLEs. Hence identification and amalgamation of engaging and easy-to-use end-user design facilities and metaphors are required. For the usability concerns, we approach a new type of mash-ups, a "flow" (see Fig. 1). Unlike dashboard like mash-ups, it tries to provide a reflection of the workflow among the widgets and the clustered nature of the learning environment.

We have elaborated on a scripting language and a design environment for realizing box like mash-ups addressing users ranging from experts to naïve [11]. Our end-user tests, particularly on the interface mockup, have revealed that the mash-up paradigm is quite new, and hard to grasp for non-experts. Developing natural and easy-to-use design environments stands as a main challenge. However, apart from appropriates of design facilities, setting a smooth balance between machine and user control is required, so that users are not overloaded or not totally dominated by the machine. In that sense, we believe that automated data linking and workflow creation, as well as adaptive recommendations are more promising than strong, rule-based adaptation.

In the light of above discussion, our proposal and expected contribution can be summarized as follows: (1) proof of concept implementation of a semantic mash-up personal and pervasive learning environment, (2) realization of technique(s) for automated generation of incomplete workflows or workflow fragments from user interactions, (3) realization of ontology based techniques for adaptive recommendations and guidance based on contextual information.

4 Implementation and Evaluation Plan

According to described challenges implementation of our proposal can be done in three stages - interface, data linking, and flow control – by following the three tiers shown in Fig. 1. First of all, the interface needs to be developed with its main features. Thus we opt for a client side realization for two main reasons: (1) to overcome performance bottleneck of a server-sided approach by shifting the PLE to the client-side, (2) to overcome authentication problems by shifting it to end-user rather than using a complicated server-sided single-sign-on approach. Once the interface is designed, the next step is the realization of a data linking infrastructure through inter-widget communication based server-sided mechanisms. Inter-widget communication will be based on a domain ontology where content and forms in each widget is annotated through embedded semantics derived from the domain ontology. Through the use of a domain ontology automated data linking between widgets can be realized without the necessity of user intervention. This advantage also applies to server-sided communication and service composition. A server-sided communication mechanism needs to be developed, as inter-widget communication is based on the actions of the learner but the content of the widgets can change due to other parties, e.g. if a friend of a learner adds new content to her blog. For this purpose, a similar approach to one presented in [12] is promising. Afterwards, a mechanism for automating the workflow in a (sub-) PLE through observing user interaction is required.

We plan to evaluate our approach along two specific use cases, each one involving a (sub-) PLE of a learner. The first one deals with a language learning environment In which a widget offers adaptive learning items (i.e., questions) to the learner, through dynamically generating the user goal with respect to the context of the other widgets available in her language learning PLE. This widget will be derived from an item-based learning environment employing our domain ontology to provide adaptive filtering and sequencing of learning items. The second scenario will cover a case where one or several digital devices are involved in the learning environment. A sound scenario needs to be described, which is comparatively difficult due to the rareness of the real life examples of Pervasive Computing. An acceptance and usability evaluation will be realized upon designed scenarios while the pedagogical evaluation of the system, which requires a substantial work, will remain as future work.

5 Related Work and Discussions

We have investigated several mash-up design and development tools (listed in [7]), in terms of their end-user facilities: (1) IBM Mashup Center, (2) Intel Mashmaker, (3) JackBe Presto, (4) Liquid Apps, (5) Open Mashup Studio, (6) Yahoo Pipes, and (7) Deri Pipes. These tools are mainly realized as box type mash-ups. They have a strong focus on content aggregation and manipulation, i.e. feeds, while providing limited support for service composition. Microformats and RDFa are not supported, and attention is given to feeds (e.g. RSS). Visual development environments are provided based on widgets, called modules or pipes. Furthermore, the underlying technologies and frameworks of these tools cannot be reviewed, as most of them are commercial products. Therefore it is not possible to compare these approaches with ours from a technical point of view.

A notable approach which is based on a concrete methodology and technology is SMashups [6]. It focuses on service composition rather than data. It follows the SAWSDL approach (Semantic Annotation for WSDL) which aims at adding semantic annotations to web services described with WSDL. A service annotation mechanism, called SA-REST, is based on Microformats [6] and RDFa [9] and used for REST-based services usually embedded in HTML pages. SA-REST and SAWSDL specify associations between the service description components and concepts in a semantic model (i.e. ontology) in order to enable semantic interoperability. A dashboard type example is given [1]; authors propose a design language model as well as visual facilities for designing and managing PLEs. Additionally a proof-of-concept is provided with the MUPPLE platform. However, the approach misses inter-widget communication, workflow generation facility, and data linking facility.

Acknowledgments. This paper is based on research funded by the Industrial Research Fund (IOF) and conducted within the IOF Knowledge platform “Harnessing collective intelligence in order to make e-learning environments adaptive” (IOF KP/07/006). Partially, it is also funded by the European Community's 7th Framework Programme (IST-FP7) under grant agreement no 231396 (ROLE project).

References

1. Wild, F., Mödritscher, F., Sigurdarson, S.E.: Designing for Change: Mash-Up Personal Learning Environments. eLearning Papers 2008(9), ISSN: 1887-1542, (2008)

2. Dillon, T., Talevski, A., Potdar, V., Chang, E.: Web of Things as a Framework for Ubiquitous Intelligence and Computing. In Zhang, D., Portmann, M., Tan, A.-H., Indulska, J. (Eds.): UIC 2009. LNCS, vol 5585, pp. 1--10, Springer, Heidelberg (2009)
3. Brusilovsky, P.: Adaptive navigation support. In: Brusilovsky, P., Kobsa, A., Neidl, W. (eds.): The Adaptive Web: Methods and Strategies of Web Personalization. LNCS, vol. 4321, pp. 263--290, Springer, Berlin (2007)
4. Sharples, M., Corlett, D., Westmancott, O.: The Design and implementation of Mobile Learning Resource. Pers. Ubiquitous Comput. 6, 220-234 (2002)
5. Krumm, J.: Ubiquitous Computing Fundamentals. Chapman&Hall/CRC, Boca Raton (2009)
6. Sheth, A.P., Gomadam, K., Lathem J.: SA-REST: Semantically Interoperable and Easier-to-Use Services and Mashups. IEEE Internet Computing 11, 91--94 (2007)
7. Taivalsaari, A.: Mashware: The future of web applications. Sun Microsystems (2009)
8. Kopecky, J., Gomadam, K., Vitvar, T.: hRESTS: An HTML Microformat for Describing RESTful Web Services. In: International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT '08), 619--625. (2008)
9. Lieberman, H., Paterno, F., Klann, M., Wulf, V.: End-User Development: An Emerging Paradigm. In H. Lieberman, F. Paterno, V. Wulf (Eds.): End-User Development. LNCS, vol. 4321, pp. 1--8. Springer, Dordrecht (2006)
10. Spiekermann, S.: User Control in Ubiquitous Computing: Design Alternatives and User Acceptance. Shaker Verlag, Aachen (2008)
11. Soylu, A., Mödritscher F., De Causmaecker, P.: Utilizing Embedded Semantics for User-driven Design of Pervasive Environments. In: The Fourth Metadata and Semantics Research Conference (MTSR), (2010)
12. Wild, F., Sigurðarson, S.E., Sobernig, S., Stahl, C., Soylu, A., Rebas, V., Górka, D., Danielewska-Tulecka, A., Tapiador, A.: An Interoperability Infrastructure for Distributed Feed Networks. In: COOPER Workshop, (2007)