Social Software Modeling and Mashup based on Actors, Activities and Assets

Evgeny Bogdanov, Christophe Salzmann,
Sandy El Helou, and Denis Gillet

Ecole Polytechnique Fédérale de Lausanne (EPFL)
CH-1015, Lausanne, Switzerland
{evgeny.bogdanov,christophe.salzmann,
sandy.elhelou,denis.gillet}@epfl.ch

Abstract. *Despite the extreme diversity of Web applications, one can find similarities among them. This paper proposes an answer to the question of knowing whether it is possible to conceptually represent different Web applications in a common manner such that they can generically be integrated in other Web applications. The 3A model developed at EPFL in the framework of the European PALETTE project is used to generalize the visual and functional properties of Web applications. A Web 2.0 personal learning environment based on the 3A model called eLogbook is used as a mashup container to integrate existing Web applications. The mapping procedure is described and illustrated with the example of an instant messaging application, showing that mashup is possible with 3A model.*

Key words: web mashups, personal learning environment, social software, activity modeling

1 Introduction

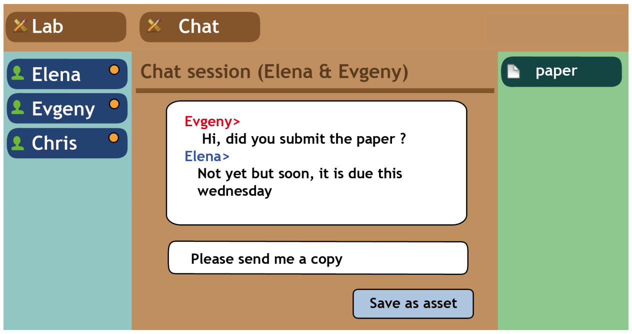
In recent years the types of Web applications has increased noticeably, from social software to online auctions and collaborative environments. The various tasks and the diverse targeted public result in numerous applications that are different in both the offered functionalities and graphical design that often make them incompatible. This incompatibility triggers the question of knowing whether there is a model to describe (some of) these applications in a common way such that they can be blended in a manner that suits users needs or wills. Some advances have been made in Web design area with commonly agreed conventions for building Web applications [1]: search fields, site id or breadcrumbs navigation become common. However, these conventions only refer to the visual aspect of a Web application, not to its functional aspect. Nevertheless, similarity among Web applications at the conceptual level can also be found and mapped into a common model, which can be used for a generic representation.

This paper is organized as follows. First, Section 2 presents the 3A model and its implementation in the personal learning environment eLogbook. Then, Section 3 depicts how the mapping of a Web application into the 3A model can be accomplished. An instant messaging mapping example and its integration in eLogbook is proposed in Section 4. Finally, Section 5 concludes and enlightens future developments.

2 eLogbook - an implementation of the 3A model

This section summarizes the 3A model description and its implementation in eLogbook that are presented in [2][3][4].

There are two well-known theories in the field of Computer Supported Cooperative Work (CSCW), activity theory [5] and distributed cognition [6]. Both theories help in learning the properties and processes of a learning system, but they do not provide concrete design specifications and cannot be directly applied for implementing a collaborative application [7]. The proposed 3A model takes its roots in activity theory, distributed cognition and actor network theory, and proposes a concrete framework for designing a collaborative web application. It consists of the three main entities Actor, Activity and Asset from which the 3A model name is derived. The main idea of the 3A model can be formulated as follows: “An Actor is producing an Asset being within an Activity”. An Actor could be a person, a software agent or any other intelligent object such as remote device. An Asset represents a document or a collection of documents or items, such as discussion thread, wiki page or image album. An Activity is the formalization of a common objective to be achieved by a group of actors. It can be the representation of a tangible space such as a classroom, or an abstract space such as a project management environment. The 3A model can be represented as a structure similar to graph. It has nodes (Actor, Activity, Asset) referred as entities connected with directed or undirected links. There can be several links between every two nodes. Each link has a specific type and weight. The richness of the proposed model also lies in the algorithms that can be used for managing and filtering the information and events related to the 3A model representation. The 3A model and its related algorithms form the core of eLogbook.



Evgeny>
Hi, did you submit the paper ?
Elena>
Not yet but soon, it is due this wednesday

Please send me a copy

Fig. 1. Instant messaging application integrated in eLogbook framework.

eLogbook is a personal learning environment that is being developed in the framework of the European PALETTE project. To facilitate the understanding of 3A model entities and interconnections several views for eLogbook are under development. Figure 1 shows the context-specific view where the surrounding elements change dynamically in function of the central focus element to represent the relation between the central element and the other 3A entities. The order in which the surrounding elements are presented is defined by various machine learned or human suggested criteria.

3 A mashup: a web application and the 3A model

A careful examination of various Web applications (chat, wiki, forums, CMS, social network, shared repository) shows that despite their difference in goals and implementations their core features can be mapped into the 3A model (Table 1). The next section shows how this mapping is performed to provide a new functional mashup enabling the representation of differently structured Web sites in a common way. A Web mashup is a Web application that combines information from two or more external online sources [8].

Web Application	Mappings
Chat	User → Actor Discussion → Space Chat history → Asset
Forum	User → Actor Thread → Space Post → Asset
Social Software Facebook, MySpace	User → Actor User profile page → Space Uploaded file → Asset
Shared Repository	User → Actor Folder → Space File → Asset

Table 1. Web Applications mapping into 3A-model

Web applications have either explicit user accounts or implicit guest/public access. Physical users are mapped into Actors in the 3A model. Besides users, Web applications have services or other autonomous entities such as recommendation mechanisms (recommender), geo-localization mechanisms or smart devices [9] that can be represented by agents and thus mapped as Actors able to perform actions. In Web applications users either work with existing content or create and manage his/her own content. These pieces of information, such as discussions, images, documents or files, are mapped into Assets. Actors and assets are usually categorized into groups or aggregated into spaces that can be mapped into Activities. Furthermore, interrelations among elements of a Web application are mapped into links among 3A model assets, actors and activities.

Personal Learning Environments (PLE) are systems that support the building of custom learning environment. These environments are defined and managed by learners to satisfy their learning needs. PLE can be composed of one or more components that communicate with each other. The eLogbook is such a PLE that relies on the 3A model to implement its internal functionalities.

By allowing the annotation and aggregation into different types and sources of information, eLogbook plays a key role in centralizing and contextualizing knowledge artifacts. Chat history files, to self-reflections, wikis, topics discussions in forums and external web links are all treated as assets and are centralized, annotated, and aggregated in the same way. The retrieval and exploitation of resources is done according to their labeling or tagging, the importance they were conferred through giving them a rate and the context in which they were placed. An example of contextualization is the linking of a chat discussion to a given topic within a community.

In addition, the way user interacts with Web applications is also harmonized. In fact, as it was mentioned above, the different resources are exploited, annotated and retrieved in a standard way, which moves the focus from learning how to use different applications to actually interacting with the artifacts themselves through eLogbook.

4 Integration of an instant messaging application into eLogbook

We implemented instant messaging (chat) in eLogbook in order to provide synchronous communication among users. Instant messaging application could easily be mapped into the 3A model. The proposed chat mashup is implemented into eLogbook as a helper application. Helper application represents the mapping of an original Web application into another (eLogbook) where not only the visual interface is translated but also the original functionalities. The mapping is performed as follows. First, the chat graphical interface is merged to the eLogbook as a dynamical page. Then, the chat model is mapped into the 3A model. The chat users become eLogbook actors and the discussion threads are held in spaces/activities. Often there is no mean to save chat discussions in standard applications. By using elogbook internal functions the proposed mashup permits to effortlessly save discussions as an asset for later retrieval. The chat functionalities are triggered when the user clicks on a specific entry in the actor list or creates a new activity with a type chat. The current and the selected actors have the possibility to instantiate and perform chat conversation respectively. Once the new activity is created, the conversation can take place. The chat administrator can use the standard eLogbook means for managing activities access rights to invite or delete a user or a group of users. If an external user does not have an eLogbook account, an email address can be used. Whenever there are new messages for a given user, he/she receives a visual notification. Clicking on the notification brings the user back to the current chat discussions. A discussion can be saved as an asset for later use by clicking on the *Save as an Asset* button (Fig. 1).

When comparing eLogbook chat implementation to the generally accepted stand-alone chat applications, the former solution shows some advantages. First, when users save discussions as assets, they can edit the saved copy to only keep the relevant parts of the discussion. The saved discussions can also be ordered (hierarchy of discussions within activities or subactivities), tagged, rated, linked and shared with other 3A model entities and/or other elogbook members. By doing this, users contribute to building their own personal learning spaces. It should be noted, that these actions are performed within eLogbook without requiring 3rd-party applications. Awareness algorithms built in eLogbook also improve the user PLE construction by providing useful information at the adequate time.

5 Conclusions and future work

In this paper we used the 3A model to represent, at the conceptual level, Web applications and links among their entities in a common way. In many Web applications one can identify one or more of the 3A model entities (Actor, Asset and Activity). The relations among these entities can be investigated and mapped into the proposed 3A model. Once the web application structure is translated to the 3A model, the visual interface and the selected functionalities can be mapped into eLogbook to obtain a mashed up application. The integration into eLogbook allows contextualizing, centralization, and annotation of different entities or pieces of information in a similar way. The centralization offered by elogbook improves information finding and discovering as well as information management thanks to the proposed annotation, tagging, rating and relating tools. A mashup example is illustrated by integrating an instant messaging Web application into eLogbook.

The proposed method is not yet automatic and efficient techniques for extracting the 3A-model structure from applications are to be developed.

References

1. Steve Krug. Don't Make Me Think! A Common Sense Approach to Web Usability. 2000.
2. Y. Rekik, D. Gillet, S. El Helou, and C. Salzmann. The eLogBook Framework: Sustaining Interaction, Collaboration, and Learning in Laboratory-Oriented CoPs. International Journal of Web-based Learning and Teaching Technologies, 2(3):61-76, 2007.
3. D. Gillet, S. El Helou, Y. Rekik, Ch. Salzmann: Context-Sensitive Awareness Services For Communities of Practice, 12th International Conference on Human-Computer Interaction (HCI2007), Beijing, 22-27 July (2007).
4. D. Gillet, S. El Helou, C. M. Yu, and C. Salzmann. Turning Web 2.0 social Software into Versatile Collaborative Learning Solutions. In The First International Conference on Advances in Computer-Human Interaction - ACHI 2008. IEEE Computer Society Press, 2008.

5. Leont'ev, A. Problems of the development of mind. English translation, Progress Press, 1981, Moscow. (Russian original 1947).
6. Hutchins. How a Cockpit Remembers Its Speeds. Cognitive Science (1995).
7. Halverson. Activity Theory and Distributed Cognition: Or What Does CSCW Need to DO with Theories?. Computer Supported Cooperative Work (CSCW) (2002).
8. Liu, Xuanzhe, Hui, Yi, Sun, Wei and Liang, Haiqi (2007). Towards service composition based on mashup. 2007 IEEE Congress on Services, 9-13 July 2007, pp. 332-339
9. Christophe Salzmann, Denis Gillet, "From online experiments to smart devices", International Journal of Online Engineering, Vol. 4, Special Issue "REV2008", 2008, http://www.online-journals.org/
10. Jonassen et al. Activity theory as a framework for designing constructivist learning environments. Educational Technology Research and Development (1999).
11. Halverson. Activity Theory and Distributed Cognition: Or What Does CSCW Need to DO with Theories?. Computer Supported Cooperative Work (CSCW) (2002).
12. Erenkrantz, J. R., Gorlick, M., Suryanarayana, G., and Taylor, R. N. (2007). From representations to computations: the evolution of Web architectures. In Proceedings of the the 6th Joint Meeting of the European Software Engineering Conference and the ACM SIGSOFT Symposium on the Foundations of Software Engineering. ESEC-FSE 07. ACM Press, New York, NY, p. 255-264.