User-aware Reactive Components for Social Networks

Benoit Baccot[1,2], Romulus Grigoras[1], and Vincent Charvillat[1]

[1] University of Toulouse, IRIT-ENSEEIHT, 2 rue Camichel, 31071 Toulouse, France
[2] Sopra Group, 1 Avenue André Marie Ampère, 31772 Colomiers, France

Abstract. Social networking Web sites are extremely popular. Users spend an increasing amount of time and expect them to be a very efficient communication medium between friends. These Web sites should be as friendly as possible, and provide users with the sense of being together. Therefore Web sites should adapt to both the way users interact with their own space or socially, with the other users. In this paper we propose to extend an existing Web adaptation platform, that allows to dynamically impact on user browsing sessions on a Web site, in two ways. First, since we apply it to social networking Web sites, we propose to extend our event model and take into account social events. Second, we propose to enrich the social networking Web sites with components that react to both the *user activity level* and the *social activity level*. The user activity level describes the low-level, basic interactions, while the social activity level characterizes the social interactions between users. Experimenting with components deployed on social networks can provide researchers with a large-scale experimental testbed.

1 Introduction

Social network sites are commonly defined (e.g. in [1]) as Web sites that have three main characteristics. First, they allow individuals to construct a public or semipublic profile, in which they can put various types of information about their personnalities, their hobbies or their status. Secondly, a list of other users with whom a given user shares a connection is also available. Finaly, the main point is that a user can use these connections to view other profile pages and go through their connection lists, to make new ones. Nowadays, social network sites, such as Facebook, Myspace or Twitter, gets bigger and bigger, and are used by millions of users a day.

The particularity of social networks is that they can provide a rich context about the users, since beyond the classic information (terminal type, available bandwidth, etc.) information related to interactions between users and content from the profile pages are available.

In this paper, we deal with this rich context and try to use an adaptation platform on social networks. This adaptation platform is able to provide Web site modifications based on the context for each users. In this paper, we want to modify a social network Web page in order to take into account the rich context, and make the site more reactive to the user.

The rest of this paper is organized in four main sections. First we present the adaptation platform, and show how it can be deployed on a social network Web site. We then define more precisely social events, that will be recorded by the platform. Two use-cases that uses the platform and the rich context available on social network Web sites are described. Some preliminary results about the implementation on Facebook and perspectives conclude this article.

2 A Dynamic Adaptation Platform for Social Networks

Generally, adaptive, dynamic multimedia applications need to take into account the context of usage in order to provide the user with a service that helps him/her accomplish a task better: find an information or a friend, communicate, play etc.

Understanding how users behave when they connect to social networking sites [2] creates opportunities for better interface design, richer studies of social interactions, improved design of content distribution systems etc. In social networks, interactions between users is paramount. These interactions raise so-called "social networking events" (here called "social events") that propagate through the network. The friends are kept posted whenever a user changes his profile, comments on another user's photo, etc. ([1]).

Interactions are mainly asynchronous: the propagation of events takes some time, and most events are generated by friends while a user is offline. Commonly the user "consumes" the events during the next session. As users are spending a large (and increasing) amount of time online, there is a need for more synchronous communication between them. Communication can be explicit (via posts, private messages etc) but also implicit. Like in real life, we may be worried if we see a friend in a sad mood. Online users may therefore be interested in knowing the state of mind of their friends at any moment. This may give them a better sense of being together ([3]).

In our previous work, we proposed a three-layer dynamic adaptation platform (figure 1) that is able to (1)observe the usage (capture various context elements), (2) analyze the usage and decide of a change to be made, and (3) apply the change. The platform naturally works in a closed-loop following the general framework of reinforcement learning: changes are evaluated with respect to some performance criteria, therefore their effectiveness can be monitored permanently. The best changes are kept, while the ones with little impact are avoided.

Also in our previous work [4] we considered the user activity level while interacting with a rich-media content. We defined the activity as the number of basic events produced by a user over a given time window. These events correspond to the basic interactions a user performs in an internet browser (e.g. mouse clicks, mouse movements, scrolls or key presses) or interactions with a multimedia component (e.g. VCR events). Considering the activity level can be useful for many applications, since adaptation actions on the Web site can be

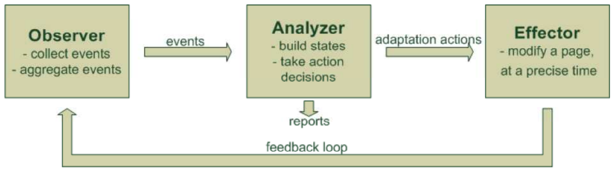

Fig. 1. The three layers of our adaptation platform.

performed upon detecting an important variation of the activity level (e.g. steep decreases or increases).

In this paper we propose to extend our dynamic platform in two ways. Firstly, since we apply it to social networking Web sites, we propose to extend our event model and take into account events at a higher semantic level: social events. Secondly, we propose to enrich the social networking Web sites with components that react to both the *user activity level* and the *social activity level*. The user activity level describes the low-level, basic interactions, while the social activity level characterizes the social interactions between users. Experimenting with our platform on a social Web site provides us with an interesting playground: once the components are deployed, and if the components enjoy a little popularity at first, the user base can increase tremendously, due to the viral nature of social networks. Indeed, this can be a very fruitful testbed on real users.

3 Social Events

Extending the event model in order to integrate social interactions is necessary for capturing user actions specific to social networks. This can be useful for both understanding (analyzing) these interaction, but also as triggers for various changes that the platform can dynamically apply to the Web site in order to adapt it to user's behaviour.

Figure 2 gives an overview of the events captured by the platform.

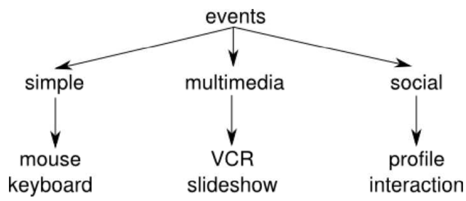

Fig. 2. The different types of events captured by our platform.

The platform is able to record simple events (events produced by mouse clicks, movements, etc.) and multimedia event (e.g. play, pause or stop on a multimedia mponent). The specifities of social networks enable us to get new events, by using the interaction logic of such Web sites. As a result, social events can be categorized in two main groups: the ones related to publishing content on his content page (e.g. publishing a picture or writing a status) and the ones concerning interactions with other users (e.g. visiting a friend's profile or commenting a friend's publication).

Social events are more complex than simple or multimedia events, since they can have multiple targets, e.g. when a new content is published on a user profile page. The cardinality of the source or destination of these events may vary.

4 Components for social Web sites

In this section, we describe two use-cases for our platform on a social network. The first one dynamically adds a component on a social network Web page, that uses the behaviour of the current user to determine its content. The second component is much more entertaining and deals with overactive users, who are proposed a slap game in order to calm down.

4.1 User-aware Contact Bar

Using the available rich context, a dynamic contact bar can be added to a social network page. This bar recommends people that have the highest interactivity level (the most numerous interactions) with the current user. By clicking on a contact name, the user can display additional information. Interactivity level between users is defined using the social activity level: in fact, when a user interacts with a contact on a social network, it generates social events that can be recorded by our platform. The user activity level is also used since this bar is more visible when the activity of the user decreases, in order to stimulate him, at a time when, for example, he could leave the Web site.

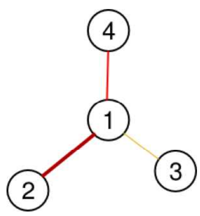

Fig. 3. The social activity of user 1. The size and the color of the branches determine the level of activity.

Social events are used to create a graph that represents the interactions between a given user and his contacts. Such a graph is presented in figure 3.

Users are the node of this graph. A link between two users means that social interactions between the users have been recorded. Links are weighted by the type and the number of times events have been captured. As a result, the graph is updated as often as social events are detected. Moreover, in order to take into account the temporal dimension, weights on link are frequently decreased. This enables to give a higher priority to recent social events, and allows the component to be more reactive to the user behaviour. Contacts to appear in the bar are then the ones with the highest interactivity with respect to the current user.

4.2 Slap Game

This use case presents a funny way to calm down overactive users, by using both the user and social activity of the user. It is really interesting to have such a funny component since it will help to disseminate the applications through through the social networks users. It also illustrates the platform ability to perform in real-time.

Generally, on a social network Web site, one can share information with other users on a profile page. Most of the time, a photo of the user is present in this page. When overactivity (in the sense of simple or multimedia events) is detected, a user is encouraged to use a slap game on a contact profile, that consists in giving slap to a contact (determined by the social activity), by moving the mouse inside and outside his contact profile picture. A score is computed, based on the velocity and the number of mouse movements. An animation is then played on the picture in order to see the effects of the slap. High scores are stored and displayed on corresponding profile pages.

5 Preliminary Results

In this section, we present the implementation of the use-cases on Facebook, using our adaptation platform. We also give some elements about the potential number of users of these components.

5.1 Deployment on Facebook

We choose to use Facebook [5] in order to implement the use cases. In fact, Facebook is unquestionably the biggest social network, with 400 millions active users who get connected in average 55 minutes a day [6]. On Facebook, the contacts of a user are called friends.

We adapt our platform (figure 1) in order to take into account social network particularities and to implement the detection of the previously called social events. We simply find rules that allow us to infer social events from simple events. Social events are of different types: chatting with a contact, adding a contact, posting a status etc.

In order to integrate it to Facebook, we had two choices:

– directly deploy the platform in the Facebook infrastructure. Obviously, as Mark Zukerberg, the Facebook creator, is not one of our acquintance, this was not a realistic solution ;).
– use a plugin for a Web browser. The plugin will be charged to dynamically add our events listeners in the Facebook Web pages, and also to add the contact bar and the slap game in the DOM tree of the Facebook pages. Then, the two components directly communicate with the platform, installed on one of our servers.

We developed a plugin for the Firefox Web browser, since installing a new plugin is quite easy (only 3 mouse clicks are needed), and the browser is used by many people. As a result, the plugin could be used by a significative number of people (see next subsection to get more details about that).

The contact bar was written in Javascript and uses the Facebook API in order to get information on the presented friends. Figure 4 presents the contact bar integrated to Facebook. The friends at the top of the bar are the one with whom the current user has the more interacted in past few minutes. It appears when a huge user activity is detected.

Fig. 4. The contact bar with two contacts.

The slap game was also written in Javascript. Figure 5 presents the slap effects on a "friend" picture profile. The slap game was proposed due to the user overactivity. As for the "friend", he was chosen using the social activity.

5.2 Dissemination of such components

Experimenting with our platform on a social Web site can provide an interesting playground. Once the components are deployed on a social network, and if the components enjoy a little popularity at first, the user base can increase tremendously, due to the viral nature of social networks. Obviously, "normal"

Fig. 5. A slapped friend and the high scores.

users needs to be interested by the components, but this is probably what may happen with the slap game. Indeed, this can be a very fruitful testbed on real users.

We present here (figure 6) the evolution of the number of users of FBOSF [7], an other Firefox plugin, that adds an "I don't care" button next to the "I like" one on Facebook friends messages, displayed on a user profile page. We have already worked with the creator of this plugin in order to deploy large scale experiments. It can be seen as a simpler component but all the same similar to the slap game, and it can enable us to think that we could have a similar dissemination for our plugin.

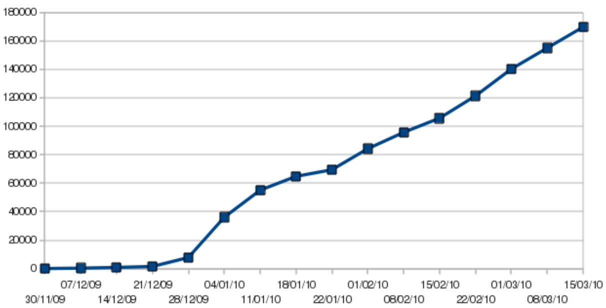

Fig. 6. Evolution of the number of users for FBOSF, 11/2009 - 03/2010.

What is really interesting is the increasing number of user (8k users after a month, 200k two months later). This encourages us to make our plugin available, in order to get as many users as possible to collect real user data.

6 Conclusion and Future Work

In this paper we presented some extensions to our original adaptation platform applied to social Web sites. In particular, we proposed novel components that can take into account both user's interactions with his own space, and also social interactions with other users.

As for the interaction graph used by our contact bar, its evolution in time (and the relation with the social graph) needs to be studied more carefully [8].

New use cases can be imagined for our platform. Examples include recommender components that suggests links according to user's behaviour, or a system that displays personalized advertisement. Statistical models such as bandit models, explored in our previous work [9], can help us optimize the performance of such components using reinforcement learning techniques that very often require a large user base. Thus, making use of the viral property of social network can enable us to test our statistical models.

7 Acknowledgments

The authors would like to thank ENSEEIHT students Anthony Luce, Guillaume Rondan, Imad El Ghalbzouri, Naoufal El Jaouhari and Simon Gay, for their participation in the development of the experimental Facebook components presented in this paper. Special thanks go to Julien Noleau, the creator of FBOSF, for his precious help.

References

1. Boyd, D., Ellison, N.B.: Social network sites: Definition, history, and scholarship. Journal of Computer-Mediated Communication **13**(1) (2007)
2. Benevenuto, F., Rodrigues, T., Cha, M., Almeida, V.: Characterizing user behavior in online social networks. In: IMC '09: Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference, ACM (2009) 49–62
3. Ellison, N.B., Steinfield, C., Lampe, C.: The benefits of facebook friends: social capital and college students' use of online social network sites. Journal of Computer-Mediated Communication **12**(4) (July 2007) 1143–1168
4. Baccot, B., Choudary, O., Grigoras, R., Charvillat, V.: On the impact of sequence and time in rich media advertising. In: Proceedings of the 17th ACM Conference on Multimedia. (2009)
5. Facebook: Social network. http://www.facebook.com
6. Facebook: Statistics page. http://www.facebook.com/press/info.php?statistics
7. Noleau, J.: Fbosf. http://fbosf.nolofone.com
8. Viswanath, B., Mislove, A., Cha, M., Gummadi, K.P.: On the evolution of user interaction in facebook. In: WOSN '09: Proceedings of the 2nd ACM workshop on Online social networks, New York, NY, USA, ACM (2009) 37–42
9. Baccot, B., Charvillat, V., Grigoras, R.: A bandit's perpective on website adaptation. In: Proceedings of the 10th Workshop on Multimedia Metadata (SeMuDaTe'09). (2009)