Supporting Offline Interaction with Web Sites Resilient to Interruptions Applied to E-learning Environments

Félix Albertos Marco

Escuela Superior de Ingeniería Informática de Albacete
Campus Universitario, 02071 Albacete, Spain
felix.albertos@uclm.es

Abstract. Despite the wide availability of Internet connections, situations of interrupted work caused by accidental loss of connectivity or by intentional offline work are very frequent. Concerned by the negative effects of interruptions in users' activities, this work investigates a new approach for the design and development of Web applications resilient to interruptions applied to e-learning environments. In order to help users to recover from interruptions whilst navigating Web sites, this paper proposes a model-based approach that combines explicit representation of end-user navigation, local information storage (i.e. Web browser caching mechanism) and polices for client-side adaptation of Web sites. With this model, we are able to provide users with information about which Web site's contents are available in an offline mode and how they can get easy access to local cache content. Moreover, the model can be also be used to set proactive mechanism such as pre-caching Web pages that are likely to be seen by users. Such model-based approach is aimed to be used to build new Web sites from scratch but it can also be used as a mapping support to describe offline navigation of existing Web site.

1 What is the Setting and History Behind This Project?

The World Wide Web has been used to consume online resources since its inception. But using the Web requires to be connected to the Internet. Unfortunately this connection is not always available: it is susceptible of been interrupted. There are many situations when users may not have or loose temporally connection to the Internet. Some examples are when traveling (e.g. by flight), natural disasters, countries in process of development or situations where the connection is not available or limited (e.g. consumption of educational resources in rural areas). Online learning environments are especially sensible to this issue. Web 2.0 technologies are making it far easier for learning communities to integrate various interests, by facilitating anytime, anywhere, anyone communication [1]. At this point, blended learning arises. There, face-to-face instruction mixes with online strategies. But there is a gap between these approaches. The transition between the classroom and student's home affects students' outcomes. Sometimes, students have to prepare material for late use. They are interrupted while changing the context between home and the class. Moreover, this interruption may be increased because the lack of Internet connection at student's home. To address this

Q.Z. Sheng and J. Kjeldskov (Eds.): ICWE 2013 Workshops, LNCS 8295, pp. 310–314, 2013.

issue are needed means to characterize offline scenarios for helping students to: 1) continue working when the interruption overcomes; 2) recover from the interruption.

2 What Is the Problem to Be Addressed?

Despite the wide availability of Internet connections, situations of interrupted work caused by accidental loss of connectivity or by intentional offline work are very frequent. The study of interruptions is relatively new and there is very little information about how interruptions affect users' activity on the Web. However, some studies in the field of Human-Computer Interaction can provide some clues about how to tackle this kind of problem. Formally speaking an interruption can be defined as a (intentional or unexpected) switch between two tasks; when an interruption occurs, users are forced to do something else (the secondary task) until the primary task could be resumed [2]. It has also been shown that interruptions will ultimately affect users' ability to complete tasks but the disruptive effect varies according to the type of interruption (e.g. system alarms and notification, deny of service, loss of connectivity...) [3]. Thus, there is no universal solution for dealing with interruptions. Nonetheless, an interruption is not a fate. Indeed, previous work [4,3,5] has shown that is possible to design interactive applications to be resilient to interruptions. The term resilient is often used to address systems that are able to recover from failures, but in the present context it is used to qualify applications that can prevent from the occurrence of interruptions, help users to resume from interrupted tasks, and/or ensure a minimum level of service for performing a task despite of the interruption [3].

Interruptions can also be very annoying when navigating Web sites that do not even require a connection because most Web browsers do not allow a natural navigation through content already stored in the local cache. Moreover, even if users are aware of the interruption, they are not able to decide what parts of the Web will be available for offline use.

Several studies have demonstrated negative effects of interruptions in users activity: resuming tasks after interruptions is difficult and can take a long time [6], interrupted tasks are perceived as harder than uninterrupted ones [7], interruptions cause more cognitive workload and they are quite often annoying and frustrating because they disrupt people from completing their work [8,7]. Interruptions can be particularly dreadful when navigating the Web because they often cause users to be disconnected from the applications, so that users should restart tasks from the beginning rather than simply resuming them.

2.1 Case Study

The University of Castilla-La Mancha has a Web site[1] where visitors can found useful information about the institution, related news and academic information among other content. For using this web, Internet connection is needed. But, if users are

[1] http://www.uclm.es

visiting the web and the connection to Internet is interrupted, they wouldn't be able of navigating visited web pages or those considered as important. They have to wait until the connection is restored.

Our proposal allows users to continue using the web according to predefined policies for dealing with interruptions. This process includes three steps. First, the structure and the behavior of the content must be described. Second, the content must be adapted when the interruption begins, following the behavior described in the previous step. At last, the content must be adapted when the interruption ends, providing the necessary mechanism for resuming previous work. This scenario allows students to continue working regardless of the connection status.

We have been working on tools for supporting the offline model. One approach is the use of a proxy[2] that: a) Allows developers and users to design the model; b) Manages interruptions and provides continuity of service when the interruption overcomes.

2.2 Objectives

Our aim is to develop a model for designing offline interaction when the connection is interrupted, providing means to: 1) help developers and users to make decisions about the content and the behaviour of the site when it is interrupted; 2) consume resources and use the web when it is interrupted; 3) recover from the interruption and resume the task normally.

The main goal is to provide a continuity of service despite of the interruption through a model for designing the offline interaction. In order to achieve this goal, the following issues have been identified: a) interruption management; b) incorporate features that support modern web applications (social, collaboration and awareness) c) design support tools to test and to evaluate the identified issues.

3 What Are Some Current Approaches to This Problem?

Disconnection is common in particular in mobile environments. For that Chang et al. [9] propose a standard browsing model that is aimed at supporting user work in disconnected mode by making the cache model transparent to both Web browsers and Web (proxy) servers. That tool contains a list of all HTML entries in the cache with a hyperlink to the corresponding contents stored locally, so it may be used for browsing local pages in offline scenarios. Other similar tools for supporting cache management are Web-Based Temwork [10] and BITSY [12]. However, all these tools don't allow tuning the Web application for working in offline mode.

Most browsers do not have mechanisms for managing Web sites in offline mode. Recently, Cannon and Wohlstadter [13] proposed a framework for offline storage that introduces automated persistence of data objects for JavaScript. Google Gears allows

[2] Available proxy video at: `http://www.youtube.com/watch?v=mglbzZwr7EU` (Last access on May 15, 2013).

browsers with the ability to persist data for offline use. However, the management of persistent data in the browser is not straightforward due to the need of synchronization, management of throughput, latency and existence of non-standards browser.

The development of Web applications supporting offline work is complex [14]. Existing applications are harder to adapt with offline support, and usually implying writing alternate versions of code [15]. Tatsubori and Suzumura [16] propose a development method that speeds up the implementation of offline work in a Web application by deploying server functionalities on the local machine. However, replicating all data to the local server isn't practical for all applications. They overcome this by enhancing the local server with an adaptive pre-fetcher mechanism that keeps fetching useful data from the remote server. Benson et al. [17] propose the synchronization of relational database between the browser and the web server and a client-side template library. Whilst that work [17] can reduce the transfer between the client and the server it does not necessarily improve the navigation into local storage.

4 Why Is This Problem Worth Solving or Worth Solving Better?

There are several attempts to formalize cognitive models describing the impact of interruptions in the human behaviour [6]. The unpredictability of interruptions would favour the use of declarative models to describe what should be accomplished by the system (whatever it happens) rather than describe the steps required (i.e. control flow) to accomplish it. Notwithstanding, there are some situations where the interruption of tasks can be predicted (in particular when users decided to get interrupted), so that the systems should provide an alternative representation of the interrupted tasks.

Only a few works in the literature have addressed the description of interruptions in system specifications [3]. It is interesting to notice that despite the fact that model-based approaches [18] are prominent in the field of Web engineering, as far we could investigate there is no clear proposal for using Model-Driven Approaches for building Web sites resilient to interruptions.

Most of Web engineering methods such as UWE [8], WebML [19], WSDM [11], OOHDM [20] and SWC [21] are useful to describe the structure of Web applications that are aimed to be deployed on Web servers and run online. The occurrence of interruptions during the execution of the Web applications is not a concern of current MDA approaches that assume that interruptions should be treated by the browser alone. As a consequence, there is no construct in such models to describe an alternative navigation for the Web application when the connectivity is lost.

References

1. Fleck, J.: Blended learning and learning communities: opportunities and challenges. Journal of Management Development 31(4), 398–411 (2012)
2. Trafton, J.G., Monk, C.A.: Task Interruptions. Reviews of Human Factors and Ergonomics 3, 111–126 (2007)

3. Palanque, P., Winckler, M., Ladry, J.-F., ter Beek, M., Faconti, G., Massink, M.: A Formal Approach Supporting the Comparative Predictive Assessment of the Interruption-Tolerance of Interactive Systems. In: ACM EICS 2009, pp. 211–220. ACM Press (2009)
4. Czerwinski, M., Horvitz, E., Wilhite, S.: A diary study of task switching and interruptions. In: CHI 2004, pp. 175–182. ACM (2004)
5. ter Beek, M.H., Faconti, G.P., Massink, M., Palanque, P.A., Winckler, M.: Resilience of Interaction Techniques to Interrupts: A Formal Model-based Approach. In: Gross, T., Gulliksen, J., Kotzé, P., Oestreicher, L., Palanque, P., Prates, R.O., Winckler, M. (eds.) INTERACT 2009. LNCS, vol. 5726, pp. 494–509. Springer, Heidelberg (2009)
6. Trafton, J.G., Altmann, E.M., Brock, D.P., Mintz, F.E.: Preparing to resume an interrupted task: Effects of prospective goal encoding and retrospective rehearsal. International Journal of Human-Computer Studies 58(5), 583–603 (2003)
7. Mark, G., Gudith, D., Klocke, U.: The cost of interrupted work: more speed and stress. In: SIGCHI 2008, pp. 107–110. ACM (2008)
8. Koch, N., Knapp, A., Zhang, G., Baumeister, H.: Uml-Based Web Engineering - An Approach Based on Standards. In: Web Engineering, pp. 157–191 (2008)
9. Chang, H., Tait, C., Cohen, N., Shapiro, M., Mastrianni, S., Floyd, R., Housel, B., Lindquist, D.: Web browsing in a wireless environment: disconnected and asynchronous operation in ARTour Web Express. In: ACM/IEEE MobiCom 1997, pp. 260–269. ACM (1997)
10. Yang, Y.: Supporting Online Web-Based Teamwork in Offline Mobile Mode Too. In: WISE 2000, vol. 1. IEEE Computer Society, Washington, DC (2000)
11. De Troyer, O., Casteleyn, S., Plessers, P.: WSDM: Web Semantics Design Method. In: Web Engineering, pp. 303–351 (2008)
12. Mehta, N., Swart, G., Divilly, C., Motivala, A.: Mobile AJAX Applications: Going Far Without the Bars. In: 2nd IEEE Workshop on Hot Topics in Web Systems and Technologies (2008)
13. Cannon, B., Wohlstadter, E.: Automated object persistence for JavaScript. In: WWW 2010, pp. 191–200. ACM (2010)
14. Gutwin, C., Graham, N., Wolfe, C., Wong, N., de Alwis, B.: Gone but not forgotten: designing for disconnection in synchronous groupware. In: CSCW 2010, pp. 179–188. ACM (2010)
15. Kao, Y.-W., Lin, C., Yang, K., Yuan, S.-M.: A Web-based, Offline-able, and Personalized Runtime Environment for executing applications on mobile devices. Comput. Stand. Interfaces 34(1), 212–224 (2012)
16. Tatsubori, M., Suzumura, T.: HTML templates that fly: a template engine approach to automated offloading from server to client. In: WWW 2009, pp. 951–960. ACM (2009)
17. Benson, E., Marcus, A., Karger, D., Madden, S.: Sync kit: a persistent client-side database caching toolkit for data intensive websites. In: WWW 2010, pp. 121–130. ACM (2010)
18. Rossi, G., Pastor, O., Schwabe, D., Olsina, L. (eds.): Web Engineering: Modelling and Implementing Web Applications. Human-Computer Interaction Series. Springer (2008)
19. Ceri, S., Brambilla, M., Fraternali, P.: The History of WebML Lessons Learned from 10 Years of Model-Driven Development of Web Applications. In: Borgida, A.T., Chaudhri, V.K., Giorgini, P., Yu, E.S. (eds.) Conceptual Modeling: Foundations and Applications. LNCS, vol. 5600, pp. 273–292. Springer, Heidelberg (2009)
20. Rossi, G., Schwabe, D.: Modeling and Implementing Web Applications with Oohdm. In: Web Engineering, pp. 109–155 (2008)
21. Winckler, M., Palanque, P.: StateWebCharts: A Formal Description Technique Dedicated to Navigation Modelling of Web Applications. In: Jorge, J.A., Jardim Nunes, N., Falcão e Cunha, J. (eds.) DSV-IS 2003. LNCS, vol. 2844, pp. 61–76. Springer, Heidelberg (2003)