WiBAF: Within Browser Adaptation Framework

Alejandro Montes García

Technische Universiteit Eindhoven
a.montes.garcia@tue.nl

Abstract. The objective of this thesis project is to develop a framework that provides mechanisms to the user to balance personalization and privacy on the web. To achieve this we suggest a browser based adaptation engine that allows the user to choose what portions of her user model will be stored on the client and what will be on the server, trading privacy for personalization. Given that the server can store some data, data-driven adaptation can be enabled, therefore a way to bring together data-driven and expert-driven adaptation has to be defined. Finally it would be interesting to perform meta-adaptation to enhance this data-driven personalization.

Keywords: Adaptation, client-side modeling, data-driven adaptation, expert-driven adaptation, meta-adaptation, server-side modeling, user modeling.

1 Introduction

The *world wide web* is an enormous hyperspace where users face the problem of information overload. Adaptive web based systems try to tackle this problem by displaying only the information that is really meaningful for the user. These systems need to collect data from the user in order to personalize the information. The set of information that the system has collected about a user is called the User Model (UM hereinafter).

UMs in adaptive web base systems are typically stored on the server. However, this has some issues such as lack of privacy, server overload, bandwidth usage, limitation of events that can be tracked, lack of context awareness, etc... To solve this problem, some client side approaches have also been proposed.

Still, client based user modeling has some other drawbacks. Typically the user has to install some piece of software, like a desktop application or a browser plugin, and techniques that rely on the comparison of several user profiles cannot be applied. P2P networks allow the analysis of several client user profiles at a time, but in that case the result will depend on the peers connected at the moment when the comparison is being performed.

This thesis project aims to balance these two approaches in a way that the advantages of both are maximized and the drawbacks minimized.

2 Research Questions and Our Approach

2.1 Balancing Privacy and Personalization

Our main research question is **how can we get the maximum profit of client side and server side modeling and at the same time the minimum inconveniences of both approaches?**

We propose to achieve this by developing a framework based on client side user modeling, in which the data will not leave users' devices. A typical issue with this approach was the need for the client to install some software that can do the user modeling and store the data, however browser storage has been enabled by HTML5, eliminating that requirement.

HTML5 storage however has some limitations as it is browser dependent, a user could have different user profiles in different devices or even in the same one if she changes the browser. This could be good when the user uses every device or browser for a different purpose, but when that is not the case, a mechanism to synchronize user profiles should be provided.

An issue remains with client side approaches to personalization, namely the inability to apply data-driven adaptation techniques. Data-driven adaptation techniques, such as collaborative filtering, typically require the analysis of several UMs at a time. Since every user only has its own UM, those techniques cannot be applied. We propose that our framework should be flexible with user choices. This means that the user could select some parts of the UM to be stored on the server instead of the client, in exchange she will get more personalization. This way we make users more aware of their privacy and what the data they submit is used for.

With such approach, the user can take care of her privacy, know what data is submitted and why it is submitted. Moreover the framework can track events (e. g. scrolling, moving the mouse) that occur while the user is browsing and are not sent to the server. This is done without any extra costs in bandwidth consumption nor server usage. Finally, context awareness is enabled so that the application can react to changes in user context.

An architecture like that brings some sub-questions such as how to optimize communication between server and client when this is required or how computationally expensive this is for mobile devices.

2.2 User Modeling

Adaptation cannot be performed without a UM. Given that we want to balance privacy and personalization at user's will, we need a language that makes it transparent for the developer where the UM is stored. We propose the development of a user modeling language that supports operations to query, store, update, delete and move data from the client to the server and vice versa.

This user modeling language will define what events will be tracked and link them to operations over the UM. It will also have to provide a uniform way to decide whether these operations will be performed on the client or on the server.

For the client operations, Javascript can be used, for the server operations we will use DimML[1]. DimML is a declarative programming language that provides an easy way to capture user events on the client side, process that data and send it to the server. It can also be used to synchronize data between devices if this is requested by the user. DimML is being developed mainly by adaptive systems expert D. Smits [2] [13] at Adversitement BV in collaboration with the Technische Universiteit Eindhoven.

Even though the language we want to define is interpreted on the client, it is sent by the server, so the server will know the structure of the UM, but not the values of the variables that the user does not want to disclose.

2.3 Further Challenges

Expert-Driven and Data-Driven Adaptation Expert-Driven and Data-Driven adaptation are two approaches that aim for personalization. To bring those two approaches together, we are developing a CSS-like language that is able to express common adaptation techniques described in the taxonomy by Knutov et al. [5]. This language has to express adaptation rules that trigger either data-driven or expert-driven procedures and allow the selection of users and contexts as shown in Listing 1.

Listing 1. Example of context and user query

```
@context (max-date: 21/03/2014) {
        @user (music-knowledge: 100) {
                p {
                        reorder-nodes: svd-recommendation;
                }
        }
}
```

With this language we achieve not only a uniform mean to express adaptation techniques regardless whether they are data-driven or expert-driven, but also a way to decouple adaptation from business logic. Decoupling these two different aspects is useful because it eases code maintenance, and compatibility with existing applications. However, this language does not deal with user modeling rules.

Given that the adaptation code is separated from the rest of the application, this code can be modified by another system in the meta-adaptation process.

Meta-Adaptation Adaptation techniques can be applied incorrectly sometimes and there are several reasons that explain why this might happen: a mistake of the domain expert that defined the adaptation rules, not having enough data to choose the proper algorithm, the discovery of new algorithms by the scientific community. That is why meta-adaptation (the adaptation of the adaptation strategy) can be useful.

[1] http://www.dimml.io/ (Last accessed on 3rd Mar 2014)

All of the rules we have introduced in the language for our framework take some parameter. The meta-adaptation process could evaluate the data available on the server and tune those parameters. After tuning them, the meta-adaptation process should check whether there is an improvement or not.

A specific example where meta-adaptation can be useful in our language is the rule that reorders nodes. This reordering follows a strategy, but it can be possible that, it is not clear what strategy would be the best. The developers can then choose one good strategy and let the meta-adaptation process change it when it has enough data to make a decision.

2.4 Summary

As a summary, this thesis project is aimed at the development of a framework that performs within-browser adaptation without the need to install any software. The most important feature of this framework is that it will balance personalization and privacy of the user in an efficient, flexible and transparent way This will be done by using a distributed user modeling mechanism that allows the exchange of information being sent to the server for personalization. This framework will also bring together two currently distinct and to a large extent isolated approaches to personalization namely, expert-driven adaptation and data-driven adaptation. This will be done by defining a language that decouples adaptation from business logic, and another one to perform user modeling. Finally, meta-adaptation can be applied by modifying the code of the aforementioned language in order to enhance the personalization.

3 Related Work

The importance of privacy in personalization systems have been pointed out by A. Kobsa [6]. Previous work to balance privacy and personalization has been done by A. Krause and E. Horvitz [9]. Their approach tries to optimize the utility-privacy trade-off by finding a subset of user features that do not allow to identify users but is significant enough to provide personalization. Our approach is different because we focus on developing transparency and giving the control to the user to select what she wants to disclose. This privacy selling idea has also been studied by C. Li et al. [10], they propose an approach in which users receive micro-payments instead of personalization in exchange for privacy loss.

Typically, adaptation was done by expert-driven systems like AHA! [2] and GALE [13]. In such systems, a domain expert defined a set of rules that would personalize the browsing experience. They usually require the development of specific applications for those frameworks, thus are not compatible with current existing web applications. Moreover, most are intended to model user features on the server side. Client side approaches have been also implemented such as the work developed by Kolias et al. [7], but it requires software to be installed.

Nowadays, data-driven systems (e. g. recommender systems) are emerging as a complement for expert-driven systems. Data-driven systems do not make use of any rule to adapt a web page, instead they use the historic data from one or more users and personalize the web page according to that data. One common example of data-driven systems are recommender systems. An excellent review of such systems has been done by B. Mobasher [11].

F. Bry and M. Kraus [1] and later M. Kraus, F. Bry and K. Kitagawa [8] propose a very inspiring adaptation language similar to the one that we pursue. They suggest that context can be stored in the DOM tree but we consider that with the HTML5 standards this is not necessary anymore. Moreover their approach is not fully compliant with all the aforementioned adaptation techniques.

In Wesomender [12] some work related to meta-adaptation has been done because the collaborative filtering component selects the best algorithm for the current data and this algorithm can change as more data is mined. Another example of meta-adaptation can be seen in Transform [3], where the adaptation rules are changed according to the battery level and how much the user participates in a video conference.

4 Progress and Future Work

So far we have done a study about how to find the equilibrium between privacy and personalization. The outcome of such study is the need for a mechanism to let the user choose what is sent to the server and what she wants to keep private. This mechanism should allow different levels of privacy that can be selected by the user. This way the user is aware of how much data she is sharing and how much personalization she is getting in exchange. However, it has to be taken into account which kind of interface should be displayed to allow the performance of this operations and to enable scrutability. This interface can also be used to disable adaptation, providing an opt-out mechanism to prohibit the user modeling process, this way our framework would be compliant with the *"Do Not Track Me Online Act"*[2]

We started the definition of a CSS-like adaptation language that will express all the adaptation techniques from Knutov's taxonomy [5]. At the same time, we are developing a Javascript interpreter for that language and a working prototype of an adaptive web system that uses it. It will allow to define user and context queries in a similar way that CSS allows to query for device properties, e. g. the size of the screen. Moreover, it defines rules that can trigger expert-driven or data-driven algorithms on the client or on the server to perform personalization.

A task we need to study is how to define a user modeling language that meets all our requirements. Moreover it is possible to tackle the cold start problem by sharing variables stored within the browser and generated by seemingly unrelated applications. This is possible by using Web Messaging [4]. However, some security restrictions have to be established so that this sharing is not done without user consent. It is also required to study how secure the browser side storage is.

[2] http://www.gpo.gov/fdsys/pkg/BILLS-112hr654ih/pdf/BILLS-112hr654ih.pdf (Last accessed on 27th Mar 2014)

Finally, another element that is worth to study is how to enable the meta-adaptation process. In Sect. 2 we have made a shallow introduction about how to achieve it, but it is worth to go deeper in this feature.

References

1. F. Bry and M. Kraus. Adaptive hypermedia made simple with HTML/XML style sheet selectors. In P. Bra, P. Brusilovsky, and R. Conejo, editors, *Adaptive Hypermedia and Adaptive Web-Based Systems*, volume 2347 of *Lecture Notes in Computer Science*, pages 472–475. Springer Berlin Heidelberg, 2002.
2. P. De Bra, D. Smits, and N. Stash. The design of AHA! In *Proceedings of the 17th Conference on Hypertext and Hypermedia*, HT '06, pages 171–195, Odense, Denmark, Aug 2006. ACM.
3. N. Gui and V. De Florio. Towards meta-adaptation support with reusable and composable adaptation components. In *Proceedings of the 6th International Conference on Self-Adaptive and Self-Organizing Systems (SASO)*, pages 49–58, Lyon, France, Sep 2012. IEEE Computer Society.
4. I. Hickson. HTML5 Web Messaging. http://www.w3.org/TR/2012/CR-webmessaging-20120501/, May 2012. Accessed: 20 Feb 2014.
5. E. Knutov, P. De Bra, and M. Pechenizkiy. AH 12 years later: a comprehensive survey of adaptive hypermedia methods and techniques. *New Review of Hypermedia and Multimedia*, 15(1):5–38, 2009.
6. A. Kobsa. Privacy-enhanced web personalization. In P. Brusilovsky, A. Kobsa, and W. Nejdl, editors, *The Adaptive Web*, volume 4321 of *Lecture Notes in Computer Science*, pages 628–670. Springer Berlin Heidelberg, 2007.
7. C. Kolias, V. Kolias, I. Anagnostopoulos, G. Kambourakis, and E. Kayafas. Enhancing user privacy in adaptive web sites with client-side user profiles. In *Proceedings of the 3rd International Workshop on Semantic Media Adaptation and Personalization*, SMAP '08, pages 170–176, Prague, Czech Republic, Dec 2008. IEEE Computer Society.
8. M. Kraus, F. Bry, and K. Kitagawa. XML document adaptation queries (XDAQ): An approach to adaptation reasoning using web query languages. In F. Bry, N. Henze, and J. Małuszyński, editors, *Principles and Practice of Semantic Web Reasoning*, volume 2901 of *Lecture Notes in Computer Science*, pages 113–127. Springer Berlin Heidelberg, 2003.
9. A. Krause and E. Horvitz. A utility-theoretic approach to privacy in online services. *Journal of Artificial Intelligence Research*, 39(1):633–662, Sep 2010.
10. C. Li, D. Y. Li, G. Miklau, and D. Suciu. A theory of pricing private data. In *Proceedings of the 16th International Conference on Database Theory*, ICDT '13, pages 33–44, Genoa, Italy, Mar 2013. ACM.
11. B. Mobasher. Data mining for web personalization. In P. Brusilovsky, A. Kobsa, and W. Nejdl, editors, *The Adaptive Web*, volume 4321 of *Lecture Notes in Computer Science*, pages 90–135. Springer Berlin Heidelberg, 2007.
12. A. Montes García, J. M. Álvarez Rodríguez, J. E. Labra Gayo, and M. Martínez Merino. Towards a journalist-based news recommendation system: The wesomender approach. *Expert Systems with Applications*, 40(17):6735 – 6741, Dec 2013.
13. D. Smits and P. De Bra. GALE: a highly extensible adaptive hypermedia engine. In *Proceedings of the 22nd Conference on Hypertext and Hypermedia*, HT '11, pages 63–72, Eindhoven, The Netherlands, Jun 2011. ACM.