

# Supporting Personalization in Legacy Web Sites Through Client-Side Adaptation

Jesús López Miján<sup>1</sup>, Irene Garrigós<sup>1(✉)</sup>, and Sergio Firmenich<sup>2,3,4</sup>

<sup>1</sup> WaKe Research, I.U. Investigación Informática, University of Alicante, Alicante, Spain  
jlm76@alu.ua.es, igarrigos@dlsi.ua.es

<sup>2</sup> LINVI, Universidad Nacional de la Patagonia San Juan Bosco, Puerto Madryn, Argentina  
sergio.firmenich@lifia.info.unlp.edu.ar

<sup>3</sup> LIFIA, Universidad Nacional de La Plata, La Plata, Argentina

<sup>4</sup> CONICET Argentina, Buenos Aires, Argentina

**Abstract.** Immersed in social and mobile Web, users are expecting personalized browsing experiences, based on their needs, goals, and preferences. However, adding personalization to an existing Web site is not a simple task for Web owners who are not personalization experts. Most of the existing personalization approaches imply extending the backend application or paying for a *personalization as a service* solution, which are more focused on improving conversion rates than on improving the user browsing experience. In this work we present a methodology to add client-based personalization to an existing Web site oriented to non-developer designers or Web sites owners. This approach allows them to define a set of personalization rules to be applied in the client-side with minimum alterations on the backend application.

## 1 Introduction

Nowadays, immersed in the social and mobile Web, users are expecting a personalized browsing experience, which adapts to their needs, goals, and preferences. This is especially important in those sites where personalization is a crucial factor and the fidelity of the user should be maintained and improved. However, adding personalization to an existing Web site is not a simple task, in fact, the cost associated to provide personalization (either by designing and implementing one or paying for PaaS - Personalization as a Service - products) could be one of the reasons behind the fact that vast Web sites do not offer any form of personalization.

From the academy, personalization was tackled from several points of views. The design of user profiles and recommendation systems were strong contributions to the field, but they require developing complex software components on the backend. In the context of model driven, there are several approaches for creating a personalized Web site from scratch [1, 3, 4]. Reverse engineering techniques [5] also allow us extracting the models from an existing Web (originally created with no models) and then applying one of these model-driven approaches to add personalization support. However, the owner of the Web site is most of the times a non-expert user, who is not knowledgeable about models and depend on others to implement this kind of solution. In this work, we present an approach for allowing Web site owners to

include personalization artefacts defined without requiring either advanced programming skills or advanced configuration. For this purpose, our contribution is a personalization framework, which provides them with a way of defining a personalization component to be injected in the final Web site. This component is formed by a set of ECA (event-condition-action) personalization rules that will act over certain parts of the DOM of the site. For building these rules we have considered the PRML (Personalization Rules Modelling Language) [3] set of events and actions. However, the approach proposes a rule-based generic engine that let us adding new kind of personalization effects.

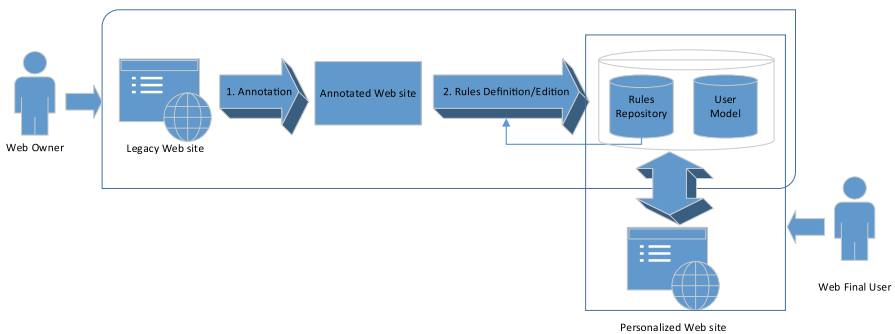
The outline of the paper is as follows. Section 2 introduces our approach from a conceptual point of view illustrated by a simple case study. Section 3 shows the implementation details, and finally, in Sect. 4 we discuss about the strength of the approach and present future work.

## 2 Personalization Support in Legacy Web Systems

In this section we present our personalization framework with the aim of giving support to non-expert users (our target are owners/designers of Web sites) in defining personalization strategies in a legacy Web system.

In Fig. 1 we can see the overall process and architecture of the framework. For demonstration purposes we have used the framework to add personalization to an existing Web site: “*La Guia X*” (<http://www.laguiax.com.ar/>), which is a shop guide of La Plata city. We describe the process next.

In order to define the elements of the Web site over which we are going to define a personalization strategy, the first step needed is the annotation of the Web site (i.e. we define the parts of the DOM where a personalization action can be defined). This step is driven by the designer/owner (using a visual tool). The framework’s extension implemented for the case study (which is shown in the following section) includes personalization strategies that act over the navigation. However, other kind of personalization strategies may be added. In the case study we considered two elements to annotate *La Guia X*: links and menus (e.g. set of links), as Fig. 2 shows. The different Web pages are also annotated with an *id* in order to refer to them in the personalization rules.



**Fig. 1.** Personalization framework overview



Fig. 2. Case study - <http://www.laguiax.com.ar/>: annotation

Once the Web site has been annotated, the next step is defining the personalization strategies to be applied. The strategies are defined as ECA (event-condition-action) rules considering the PRML language set of events and actions [3], as Fig. 3 shows. In this example, we want to sort the set of links with *id* = “Rubros”, the personalization strategy we want to define is: “sort the set of links by use, where the most used links are on the top of the list”. For this purpose, we select the action “SortLink”. Once selected, we have to define the event and condition for applying this action.

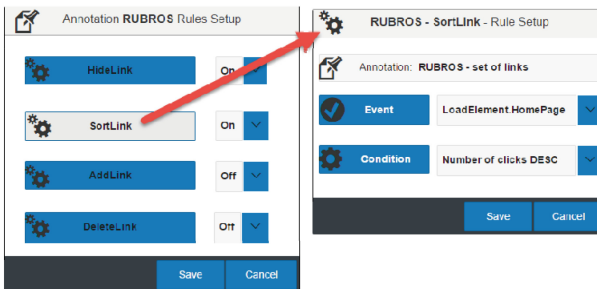


Fig. 3. Rules attached to annotation “Rubros” and “SortLink” personalization rule details

The possible set of events that we consider is the one defined in PRML lite (i.e. *LoadElement*, *Navigation*, *SessionStart* and *SessionEnd*). In this case we have set the *LoadElement* event over the home page, so every time we load the home page of the site the rule is evaluated. The condition will evaluate the usage of the links in order to properly sort them. For this purpose, we check the number of clicks done by the user of every annotated link we want to sort. This information is stored for each user in the client-side (i.e. in the local storage of the browser). An algorithm compares the number of clicks sorting them in descending way.

As we explain in the following section, the defined rules are stored in a script, which is integrated in the Web site. Although in this paper we have implemented the PRML set of events and actions, new rules can be easily added. In spite of the concrete rules, the main contribution is a little architecture for adding personalization in legacy Web sites, which are not maintained by developer teams, but still could improve the user experience with adaptation mechanisms.

### 3 Implementation

The implementation of our approach is composed by two components. The first one is an authoring tool (based on visual programming) that allows Web site owners to annotate and define the rules, and a rules engine that must be embedded in the Web site in order to execute the rules when a Web page is loaded on the client-side.

The authoring tool is implemented using a client-side plugin that allow Web owners/designers to create and edit the rules. It is implemented with common client-side Web technologies such as JavaScript, HTML5 and CSS. At the end of the definition process, this tool produces a minimized JavaScript artefact with the specification of the personalization strategy. These specifications should be added to the Web site (just by adding the JavaScript file in those Web pages where the personalization will work). Once added these specifications together with the rule engine JavaScript library, the personalization effects can be performed on the client-side. In this way, any end-user who visits the Web site has the new mechanism available.

### 4 Conclusions and Future Work

Personalization is a powerful mechanism to improve the user experience, but it is not broadly implemented in other domains than e-commerce. However, users are really interested on making adaptive the Web sites they use as shown by some trends such as Web Augmentation, even more, they can do it with end-user programming tools [2]. However, this kind of tools do not reach the whole amount of Web site visitors and do not allow Web owners taking advantage of tracking the user behavior. In this work, we use client-side adaptation to provide Web site owners with a framework for adding personalization rules to their Web site with minimum efforts and little technical knowledge. Then the rules are added to the Web site and consequently available for every visitor. As future work, we will work on an assisted annotation tool for abstracting DOM elements into business domain objects, which will allow to define more complex personalization effects such as item recommendations, etc. We are currently developing a server-side component for sharing data among devices, which also will allow deploying our approach as a third-party application included on the Web sites, in the same way than common social widgets or analytics. Moreover, we plan to do usability tests in order to improve the Web owners experience.

## References

1. De Troyer, O., Casteleyn, S., Plessers, P.: WSDM: web semantics design method. In: Rossi, G., Pastor, O., Schwabe, D., Olsina, L. (eds.) *Web Engineering*, pp. 303–351. Springer, London (2008)
2. Díaz, O., Arellano, C., Aldalur, I., Medina, H., Firmenich, S.: End-user browser-side modification of web pages. In: Benatallah, B., Bestavros, A., Manolopoulos, Y., Vakali, A., Zhang, Y. (eds.) *WISE 2014, Part I. LNCS*, vol. 8786, pp. 293–307. Springer, Heidelberg (2014)
3. Garrigós, I., Gómez, J., Houben, G.-J.: Specification of personalization in web application design. *Inf. Softw. Technol.* **52**(9), 991–1010 (2010)
4. Koch, N., Kraus, A., Zhang, G., Baumeister, H.: UML-based web engineering - an approach based on standards. In: Rossi, G., Pastor, O., Schwabe, D., Olsina, L. (eds.) *Web Engineering*, pp. 157–191. Springer, London (2008)
5. Martin, A., Cechich, A.: A model-driven reengineering approach to web site personalization. In: *Proceedings of the Third Latin American Web Congress, LAWEB*, p. 14. IEEE Computer Society, Washington, DC (2005)