Automatic Web Design Refinements based on Collective User Behavior

Luis A. Leiva
ITI/DSIC, Universitat Politècnica de València
Camí de Vera, s/n – 46022 Valencia (Spain)
llt@acm.org

CHI'12, May 5–10, 2012, Austin, Texas, USA.
ACM 978-1-4503-1016-1/12/05.

Abstract
This paper explores the following question: Could web browsing data be used to inform design refinements? An interactive tool to help website builders in the process of redesigning web layouts is introduced. The novelty of the approach is that visual modifications are generated, either completely or partially unsupervised, according to the collective behavior of the website visitors. Implications of the method and its importance for the HCI community are discussed as well.

Author Keywords
UI Redesign; Crowd Behavior; Implicit Interaction

ACM Classification Keywords
H.5.2 [**User Interfaces**]: Screen design; H.5.3 [**Group and Organization Interfaces**]: Web-based interaction

Introduction
Prototyping tools are commonplace in web design, being often used to achieve a concrete decision or to discard non-viable proposals, if any. The primary purpose of these tools is to provide feedback to define a design earlier, when there is inadequate information to choose one solution over another. However, once a website design leaves the testing phase and moves to production, it hardly ever gets substantially modified. Rather, it follows

a cycle of subtle iterative improvements. At this stage, surprisingly, a few methods seldom support incrementally revisiting different versions of the *same solution*.

This paper retakes the follow-up questions thrown by Tohidi *et al.* [10] when "getting the right design and the design right". Typically, research has looked at crafting prototypes that reflect fundamentally different designs to later compare; but, What about the details of a particular design? and, more importantly, How might these details impact what comes next, as we pursue that particular design? This work is inspired by these considerations.

Designing Alternatives != Refining Design
Much work has been done in generating design *alternatives* to assist the user in the design process, i.e., to get the "right design". For instance, Design Gallery [6] created a broad selection, automatically generated and organized, of perceptually different graphics, e.g., changes in light selection and placement of objects in a scene. Side Views [9] presented the user with real-time rendering of possible interface changes; so that the designer could see how the final design would look like, before committing any change to the UI. Finally, Adaptive Ideas [4] aimed to facilitate design by example modification, browsing and borrowing from a corpus of structured web templates.

As noticed, the notion of having the computer actively assist the designer is appealing. However, there is little research towards tools that allow designers to explore design *refinements*, i.e., to get the "design right". For instance, Ivory and Hearst [3] employed learned statistical profiles of good websites to suggest improvements to existing designs; however, changes would be manually implemented. What would be interesting, though, is being able to automate the process to a greater or a lesser extent. In this regard, Masson *et al.* [7] developed Magellan, a system that aims to foster creativity by adding permutations to an existing design. Using interactive genetic algorithms, Magellan creates and makes "evolve" a population of UI transformations. This approach relies on a user-task model and therefore it must be learned. In contrast, the method presented in this paper is a model-free approach based on a pragmatic concept: let *all* the users take part in the design process. As described later, the system modifies 'visual weights' of page elements: color, contrast, size, alignment, etc. using client-side browsing data alone (e.g., mouse movements, clicks, or key strokes); in order to generate designs that affect those elements that users interacted with most (or less), in terms of captured UI events.

Method Rationale

It has been stated for a long time that we can expect limited results from explicit user feedback [1], and so it is important to consider some alternatives. Moreover, the Web provides unprecedented opportunities to gather real data from real users, and much has been speculated about how implicit page-level interactions may improve web interfaces (e.g., see [2, 5]). Concretely, this paper follows the method described in [5], i.e., overriding the cascading style sheets (CSS) according to the frequency of users' interactions. However, instead of adapting a UI to an individual, here all client-side interactions are taken into account to alter the design of the whole website. Among other benefits, automatically translating user interactions to CSS transformations may allow designers to:

- avoid having to recruit users for testing each time the website is updated: what you see is what users do;
- discover visually what behavioral patterns are consensus;
- find inspirational examples, by looking at how the appearance of the site gets modified over time.

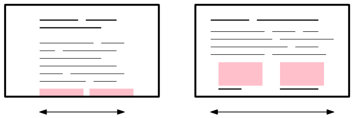


Figure 1: Widening the central column allows the browser to display more information at a glance.

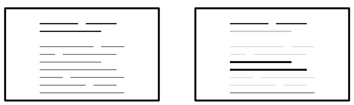


Figure 2: Some contents of the page can be altered according to its importance; e.g., changing the font sizes and colors of headings and text paragraphs.

If subtle design modifications are needed to refine an existing layout—as it often happens when iterating over a design solution—then implicit user interaction can be valuable to this end. For instance, if all users spend most of their browsing time on the home page 'above the fold', the designer could consider make wider the main body content, so that some parts could be accessed faster (Figure 1). Similarly, if there is some paragraph that is commonly selected, if would be interesting to make such text more prominent, probably by increasing the font size or the color contrast, so that in subsequent visits users could realize easily where is the popular information (Figure 2).

I advocate therefore to explore the collective users' behavior as an inspirational source for web redesign. Page-level interactions can be gathered at scale on a daily basis, and without burdening the user: independent feedback is received from hundreds or thousands of remote anonymous users rather than being produced and interpreted in a small group or individuals working in isolation. This may help to achieve (hopefully) better design decisions, since it is possible to empirically validate how users react to a particular design update; e.g., by carrying out A/B tests. Additionally, this has the notable advantage that data acquisition and later processing can be both completely automated. With these criteria in mind, an interactive tool was developed.

System Description

The tool is composed of three parts: a client-side tracking script, a server-side data processing module, and a visualization application that runs on a browser.

Firstly, a javascript program captures in the background a series of browser events (e.g., `mouseover`, `click`, or `keypress`). Since events are related to UI elements (objects) of the document object model (DOM), the program assigns a score to each DOM object based on the frequency of these captured events. Interaction data are then logged in a database in the form of 'events dictionary', where keys are DOM fragments in XPath notation and values are computed scores. Mouse coordinates are also recorded, for visualization purposes.

Later, a server-side application computes a single score for each interacted element, taking into account all the aggregated data. This way, the importance (or saliency) of a DOM object is determined by its score: the more the users interact with an element, the stronger the relative importance regarding the rest of interacted elements and vice versa. This assumption has been backed up in the literature [5, 8], and indeed it has been shown to perform reasonably well in a previously implemented prototype [5]. Scores are bound to the interval $(-1, 1)$, so that CSS values cannot be significantly altered: e.g., a score of 0.5 means "increase by 50%".

Scores are then used to modify the CSS properties of the DOM objects. Since the system is operating with implicit user feedback, it can incorporate optional human supervision. This way, non-consistent redesigns can be minimized or avoided at all. For instance, many objects of little interest are likely to be hovered when moving the mouse to a specific location and hence they would be taken into account for scoring in a fully unsupervised way. For that reason, the designer can gain control over *which* elements are going to be affected by the redesign proposals and *how*, by means of a control panel (see Figure 3) in which constraints are specified in CSS-based syntax, so that no special skills are needed to operate it.

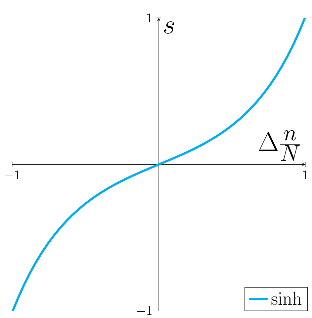
Framework Overview

CSS overriding
The value v of a CSS property is modified according to [5]:

$$v = v\left[1 + \sum_{i=1}^{m} w_i \, s_i\right]$$

where s_i is a score and w_i is a weight for each of the m types of events (e.g., `click` or `keydown`) s.t. $\sum w_i = 1$.

Non-linear scores
A score s_i is proportional to the number of events of type i (n_i) and the total number of gathered events (N) between browsing sessions [5]:



Scoring fusion
Final scores are averaged for all considered logs.

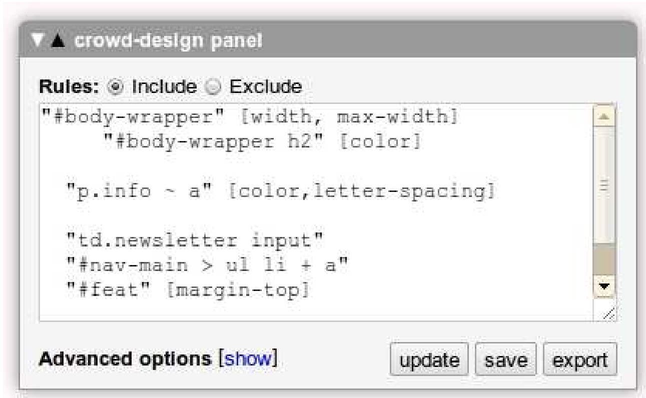


Figure 3: Designers can control which elements can be affected by the automatic refinements, by using CSS selectors notation in quotes, and specify a subset of each element's CSS properties in CSV format in brackets, so that not all modifications affect all styling. Advanced options allow, e.g., to tune the time span of gathered log files that should be processed. Redesigns can be exported to a CSS file.

Discussion

When people is exposed to different design prototypes they provide more suggestions for improvement compared to those who only see one [10]. This is useful for "getting the right design". Nonetheless, by being exposed to the automatic design refinements provided by this tool, designers would know what aspects could be improved before proceeding with "getting the design right".

Previous informal meetings with web designers have shown that this tool is perceived as a useful help. People commented that they often want to determine how changes to a few page elements will affect the final appearance of the website. This tool satisfies this need, by letting them to inspect how the users' behavior would influence the CSS rendering. Moreover, automatic redesign frees the web designer from the need to know what changes are possible, or how they can be effectively performed. Also, design refinements can offer pragmatic value as well as inspirational value. Figure 4 depicts some examples that this tool can produce.

The tool has some implications for participatory design as well, since it aims to create websites that are more appropriate to their users. It also allows designers to find "interaction agreements" between all website visitors; which may be useful to detect whether if a design works as expected, e.g., how designs change through time according to the heterogeneous behavior of the users. Additionally, non-experienced designers can gain insight about what is going on with their designs, from the user interactions' point of view. This is an aspect of design practice from which the HCI community may well be able to benefit.

Finally, collected data can be reused to support design decision making, or to improve understanding of how users interact at scale. Data can also be used for complementary analytics in traditional usability tests, or applied to infer new knowledge for future pageviews.

Limitations

First, an inherent limitation of the method, due to its automated nature, is that it cannot modify non-numerical CSS properties (e.g, the value of `text-transform` or `display` is a string). Second, it is clear that complex redesigns cannot be performed with simple changes to the style sheets. For example, while there is a positive benefit for automated modifications, sometimes it is also necessary to re-arrange several UI components beyond alignment or sizing, which would require a technically more sophisticated approach.

(a) Original design **(b)** Heatmap of Movements+Clicks **(c)** Redesign suggestion #1 **(d)** Redesign suggestion #2

Figure 4: Redesign examples produced by this tool taking into account near 100 interaction logs, after inserting a few custom rules in the control panel (Figure 3).

Finally, this method is centered around interaction data, without taking into account the semantic of the elements, and therefore it does not provide definite design solutions. If a design is weak, then the system may not be able to derive useful suggestions to "optimize" it. Instead, as previously discussed, it offers recommendations about subtle visual design modifications.

It is worth pointing out that the visual refinements are just *what-if* suggestions that leave unchanged the original CSS and HTML code of the website. Therefore, this method is a complementary medium, and so designers can later incorporate the refinements to the production version of the website or just use them to create other variations.

Conclusion and Future Work

Automatically mining crowd-based behavior for UI redesign is a promising direction for future research. As such, some work still remains to be done. Mainly a series of user studies are needed to assess in-depth the utility of the tool. I also plan to incorporate other features that would enhance the redesign process further. For instance, combined with the classical server logs information, designers would be able to filter population segments such as country, operating system, user agent, and so on.

Future work will consider coupling user interaction logs with other types of information, such as eye-gaze or other biometric data, so that redesigns can be informed by richer HCI signals. Further research will pursue more ambitious results, such as inferring high-level behaviors from low-level events — for instance, reporting if a certain design causes users to get lost or incites them to being more active.

Acknowledgements

Work supported by the Spanish research programme Consolider Ingenio 2010: MIPRCV (CSD2007-00018) and CasMaCat Project 287576 (FP7 ICT-2011.4.2).

References

[1] Hill, W. C., and Terveen, L. G. Involving remote users in continuous design of web content. In *Proc. DIS* (1997).

[2] Huang, J. On the value of page-level interactions in web search. In *Proc. HCIR* (2011).

[3] Ivory, M. Y., and Hearst, M. A. Statistical profiles of highly-rated web sites. In *Proc. CHI* (2002), 367–374.

[4] Lee, B., Srivastava, S., Kumar, R., Brafman, R., and Klemmer, S. R. Designing with interactive example galleries. In *Proc. CHI* (2010), 2257–2266.

[5] Leiva, L. A. Restyling website design via touch-based interactions. In *Proc. mobileHCI* (2011), 599–604.

[6] Marks, J., Andalman, B., Beardsley, P. A., Freeman, W., Gibson, S., Hodgins, J., Kang, T., Mirtich, B., Pfister, H., Ruml, W., Ryall, K., Seims, J., and Shieber, S. Design galleries: A general approach to setting parameters for computer graphics and animation. In *Proc. SIGGRAPH* (1997), 389–400.

[7] Masson, D., Demeure, A., and Calvary, G. Magellan, an evolutionary system to foster user interface design creativity. In *Proc. EICS* (2010), 87–92.

[8] Shapira, B., Taieb-Maimon, M., and Moskowitz, A. Study of the usefulness of known and new implicit indicators and their optimal combination for accurate inference of users interests. In *Proc. of SAC* (2006), 1118–1119.

[9] Terry, M., and Mynatt, E. D. Side views: Persistent, on-demand previews for open-ended tasks. In *Proc. UIST* (2002), 71–80.

[10] Tohidi, M., Buxton, W., Baecker, R., and Sellen, A. Getting the right design and the design right: Testing many is better than one. In *Proc. CHI* (2006), 1243–1252.