Assisting interface for easy administration of a CMS

Thibault Raffaillac
Kungliga Tekniska Högskolan
traf@kth.se

Abstract
In this paper I describe the design of an embodied assistant tool meant to ease the creation and daily administration of a website.

Keywords
Website, CMS, administration, embodied assistant

Introduction
Today, creating a website is a complex task. While hand-coding still allows people to create websites of a few pages, Content Management Systems (CMSs) become indispensable for large-scale websites. These tools are template websites that offer an interface to modify them online: no more coding, everything is managed through an Administration section.

However, CMSs still have one inconvenience: it requires a lot of knowledge to create and maintain such a website. One has to care about security, design rules, Search Engine Optimisation (SEO), charsets, mobile devices, to cite only a few. This is almost impossible and often results in flaws in the websites.

Final project in Interaction Design. DH2626. KTH, June 2011

In practice, a preliminary study showed that the main existing CMSs relate no specific feature to ease their own administration.

CMS	Woorank	How easy is the administration?
Wordpress	85,8	Few key features
Drupal	85,7	Bloated
Joomla	79,9	Nice menus & buttons
Moodle	78,8	Small ambition, so seems so
MediaWiki	77,1	Same limited ambition
Mambo	75,1	Not mentioned
MovableType	71,4	Partial ambition, so seems a bit so
Liferay	70,8	Multiple admin features (maybe too much)

Figure 1: Comparison of top-rated [7] CMSs.

In this context, I considered introducing an embodied assistant, to turn the CMS from a bunch of options to a proactive interface. This article thus focuses on two questions: Can an embodied assistant ease the administration, and how to shape it?

There are however a few limitations to this. As artificial intelligence might be hard to achieve, the messages from the assistant are limited to pre-made text. Additionally, they must be carefully composed [5], since administrators have limited will power and time, and do not want to be considered as idiots.

What are the goals of the assistant?

I started by enumerating the different aspects of the administration of a CMS, those that the assistant should help improve. For each of these, statistics would be monitored, triggering the assistant to make observations [1] if necessary:

- Clarity of the textual content
- Speed (the delays for page loads)
- Browsing for mobile devices or disabled people
- Addictivity (the user should stay longer on the website, and come back often)
- Beauty of the layout
- Content contributed from users
- Law-conformance and content licensing
- Structured / aggregated / summed content
- Dated content
- Public / protected content
- Geolocalization (deliver different pages depending on the user's region)
- Monetization (managing internal and external ads)
- Social targeting (also includes protection against inappropriate content)

A series of interviews was then conducted to check the users' receptivities to an embodied interface, and how they would imagine it. Surprisingly, they showed high expectancies regarding such an assistant: they want to be taken by the hand and guided, and they do not seem to need any control over the assistant. One of the interviewees answered the question "What would you ask him?" (the assistant) by "Teach me everything you have.".

The second observation after the interviews was that the users need to associate the assistant with a personality: it could borrow the image of someone famous, but above all its messages must conform to a defined coherent personality, and not seem like a collection of neutral advice.

How to deliver the messages?

Let us first analyze two examples of unsuccessful and successful similar assistants.

Microsoft's paperclip

This assistant from Microsoft Office was intended to help users perform usual operations, but received strong negative feedback. Three reasons can explain its unsuccess. The first is that the advices given are too basic and often synonymous to reading the software documentation. The second reason is the cartoon aspect of the assistant, treating users as kids. Finally, the third reason is that the assistant often seemed to be popping out for no reason.

Power Chess's Queen

The game Power Chess (Sierra, 1996) featured two characters: a Queen and a King. The user would first play against the King. Then, the Queen would analyze the game, arguing about how good each move was, and telling what she would have played instead. Three reasons also explain its success. The first is that her analysis was technical and relevant, and did not make trivial remarks. The second reason is the human face, assisted by recorded voice and a warm tone. The third reason is that it would also tell when a move was good, and greet the player accordingly.

According to those observations, I gave a human, serious look to the assistant. As an example character I chose a chess player [6]. A little humor may be used, but sparingly. The observations are not trivial and always link to one or two valuable sources [2]. The assistant would sometimes greet the administrator when it notices one of the administration's aspects has been well managed. Finally, the assistant occurrence should be set by the administrator, to avoid the impression of "popping out".

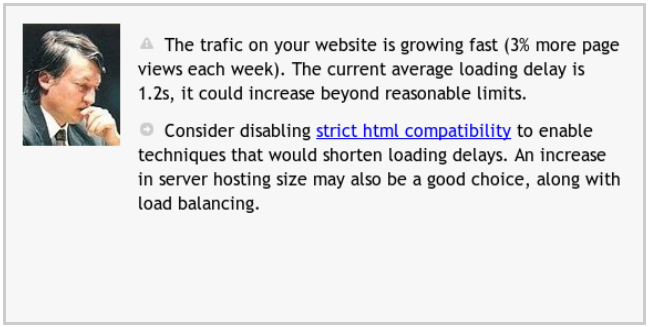

Figure 2: Sample from the prototype [4].

When to trigger the assistant?

This part deals about when and where the assistant should appear. I identified two possible approaches:

- Notification delivery, with messages appearing when the user is most inclined to handle them [3]. To forbid any pop-out window, the messages can only appear at page loads.
- Pending messages, as in all webmail applications, where an image or bold font indicates that unread messages are pending.

However, the first approach may be ill-implemented, as guessing a proper triggering moment is hard [3]. It could then become annoying for the user to see the assistant appear without knowing *why* it appeared. I thus adopted the second approach, with the exception that only one message can be pending, the frequency of a new message being limited to one or two days. This is to insist on the fact that *every* message is important and should be granted a few dozen minutes.

Conclusion

Along with this paper comes a prototype [4] which contains examples of messages that could be delivered by the assistant. As showed in the preliminary study of the main CMSs, the introduction of an administrator assistant would be a unique differentiator for any CMS implementing it, if well enough designed.

Due to time constraints, this study had inherent limitations that could be addressed in further work. First, despite being based on successful and unsuccessful designs, the prototype has undergone too little feedback from experts and users. Second, the assistant messages remain to be written for each aspect enumerated in the first part, though a few examples are provided in the prototype. Third, voice samples were judged as valuable idea but not tested, and I think it might be worth studying. Last but not least, notification delivery was discarded due to the risk of ill-implementing it, but it could actually be introduced, with a lot of care and testing.

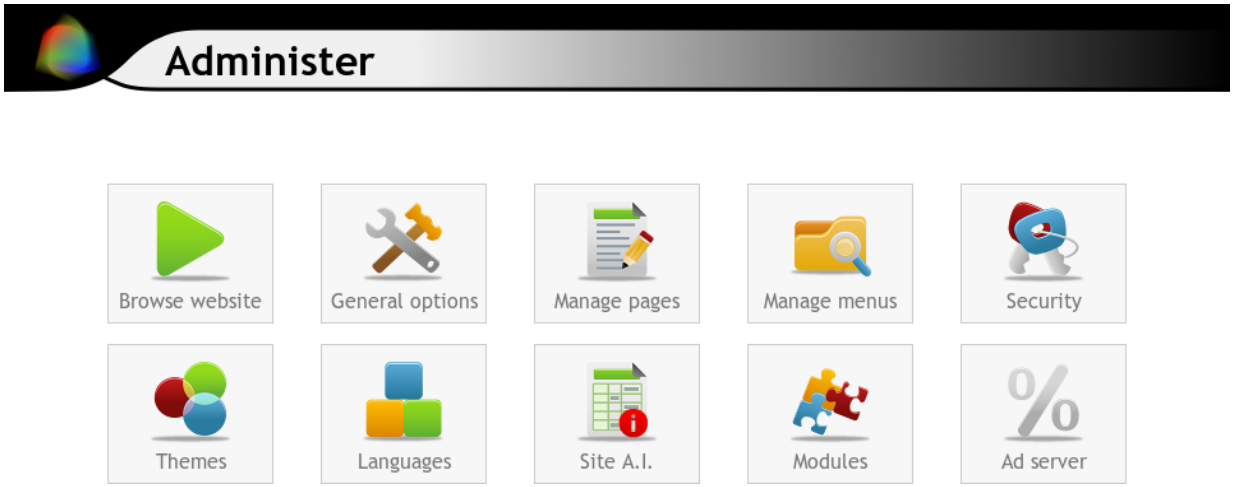

Figure 3: The prototype administration homepage [4]. The Site A.I. icon shows that a message is pending.

[1] Bhavnani, S.K., Peck, F.A., Reif, F. Strategy-Based Instruction: Lessons Learned in Teaching the Effective and Efficient Use of Computer Applications. *TOCHI 2008*, ACM Press (2008).

[2] De Bruijn, O., Spence, R. A New Framework for Theory-Based Interaction Design Applied to Serendipitous Information Retrieval. *TOCHI 2008*, ACM Press (2008).

[3] Iqbal, S.T., Bailey, B.P. Oasis: A Framework for Linking Notification Delivery to the Perceptual Structure of Goal-Directed Tasks. *TOCHI 2010*, ACM Press (2010).

[4] Online prototype for the assisting interface. http://www.csc.kth.se/~traf/cms_proto/admin.html#.

[5] Sharit, J., Hernández, M.A. and Czaja, S.J., Pirolli, P. Investigating the Roles of Knowledge and Cognitive Abilities in Older Adult Information Seeking on the Web. *TOCHI 2008*, ACM Press (2008).

[6] Van Vugt, H.C., Bailenson, J.N., Hoorn, J.F. and Konijn, E.A. Effects of Facial Similarity on User Responses to Embodied Agents. *TOCHI 2010*, ACM Press (2010).

[7] Website analysis tool. http://www.woorank.com/.