
Active Progress Bars: Facilitating the Switch to Temporary Activities

Christophe Hurter

Civil Aviation Research Center,
IRIT Toulouse University
7 Avenue Edouard Belin
31055 Toulouse cedex, France
christophe.hurter@aviation-
civile.gouv.fr

Audrey Girouard

Queen's University
Human Media Lab
25 Union St
Kingston, ON K7L 3N6, Canada
audrey@cs.queensu.ca

Nathalie Riche

Microsoft Research
One Microsoft Way
Redmond, WA 98052, USA
nath@microsoft.com

Catherine Plaisant

University of Maryland
Human Computer Interaction Lab
2117C Hornbake South Wing
University of Maryland
College Park, MD 20742, USA
plaisant@cs.umd.edu

Abstract

In this paper, we seek to find a better way of effective task management when a progress bar interrupts user's primary activity. We propose to augment progress bars with user controlled functionalities facilitating the switch to temporary activities. We detail a taxonomy of waiting period contexts and possible temporary tasks, then report on 5 participatory design, and a follow-up survey of 96 respondents. Finally we describe an early prototype of active progress bars, and report on initial use.

Author Keywords

Progress bars, participatory design, frustration, task switching.

ACM Classification Keywords

H5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

Introduction

Computer users are impatient [3]; they want to minimize their waiting periods (booting, software launching, files downloading...) when using their computer devices. Many designs have been proposed to improve the progress bars themselves [4] [5], but little has been done to help users make better use of the waiting time. The mobile applications on smart phones have clearly demonstrated that users can accomplish many short

Copyright is held by the author/owner(s).

CHI 2011, May 7–12, 2011, Vancouver, BC, Canada.

ACM 978-1-4503-0268-5/11/05.



Figure 1: When asked to wait users are often frustrated. Could productive or relaxing activities be proposed?

tasks in a just a few minutes—even seconds—which can be productive or entertaining. We believe progress bars can be augmented with simple functionalities facilitating a switch to a temporary activity, and a smooth return to the primary activity. The remarkable adaptability of computer users means that users develop their own strategies. They may stretch, head for the coffee pot, rest (Figure 1), or stay on the computer and manually switch to a temporary activity such as checking email or Facebook, or reviewing their to-do list. Sometimes they may forget to return to the primary task or choose not to return, breaking the flow [1].

The primary activity, its context and the duration of the wait may determine the type of temporary activities users engage in while the primary one is on hold. Context will also dictate the benefit – or danger - of switching to another activity. Users trying to stay focused on a single task may not want to switch, while others may welcome assistance in switching between tasks. Our goal is to investigate user interface designs that would facilitate task switching when users welcome it. We hypothesized there would be significant user differences in the choice of temporary activity and the level of control needed, and hoped to identify key design elements that would make active progress bars more beneficial.

Previous work investigated the complex research topic of self-interruption [6]. It must be noted that task switching comes at a cost **Erreur ! Source du renvoi introuvable.** Researchers keep trying to better understand the effects of interruptions which are typically considered a significant nuisance (e.g. [2]). Interruptions are usually described as something which is not under users' control (e.g. somebody walking in the office), while a progress bar is generally the result of us-

ers' actions and may be considered a break and not an interruption.

In this paper we propose the concept of Active Progress Bars to facilitate the switch to temporary activities. We first describe a taxonomy of waiting period contexts by discussing characteristics of primary and temporary activities, and waiting periods. We then report on 5 participatory design sessions, and a follow-up survey of 96 respondents. Finally we describe an early prototype of active progress bars, and report on initial use.

Characterizing Waiting and Progress Bars

To guide our exploration of active toolbars we first defined a set of dimensions describing waiting and progress bars. A simple taxonomy was defined, and then iteratively refined as we analyzed the data from our participatory design sessions.

We identified three main temporal components: (1) the primary activity the user is performing before the progress bar appears; (2) the waiting period during which the user is shown a progress bar; and (3) the temporary activity the user can perform while waiting. For each component, we identified multiple dimensions.

Primary Activity – Context

We identify three types of contexts: *work*, *entertainment* activity, and *idle*. We use context as a dimension and not the software currently used, as this might vary per user. For instance, checking emails might be considered work if done by a lawyer during a week day, but might be entertainment for a teen user. The third context – *idle* - refers to situations when the user was not performing any specific task before the start of the progress bar (such as when booting up a computer).

Primary Activity – Stage

When the waiting occurs, the primary activity can be in one of four stages: (1) at the start of the activity, when loading a program; (2) during the activity, when performing data transformation; (3) at the end of the activity, when logging off; and finally (4) in a peripheral activity, when downloading a file required later.

Waiting period – Control of initiation

The start of the waiting period can be under user control or not. Users decide when to compile their code or save a large document, but the waiting may also be triggered automatically during the primary activity, such as when a large document is saved automatically.

Waiting period – Duration predictability

The waiting period can be either *determined* or *undetermined*. Determined waiting periods correspond to situations where an algorithm can estimate the length of the wait. Often the duration of the wait cannot be determined. It is often expressed in the interface by a rotating wheel or hourglass instead of a progress bar.

Waiting – Duration

The duration of waiting period can vary significantly. To simplify, we distinguish four ranges: the waiting can be *instant* - less than 1 second of wait, *short* - less than 5 seconds, *medium* – between 5 and 15 seconds, or *long* - more than 15 seconds to wait.

In this paper, we will discuss short, medium and long waits because they can be perceived by the user and influence the decision about what to do during this period. Note that recent work on progress bars only focused on short or medium wait times (5s for [4] [5]).

Waiting – Notification

We observed three types of notification of the progress during the waiting period. It can be discrete (e.g. number of seconds, if the end of the wait time is emphasized), continuous (percentage done), binary (icon or message) or there can be no notification at all.

Temporary Activity – Context

The temporary activity has a context with the same characteristics of the primary activity (work, entertainment or idle). However, the context of the temporary activity does not need to be identical to that of the initial activity. One could watch a short fun video (entertainment) while installing a statistical package (work). The idle context of a temporary activity refers to the user stopping all computer activity, e.g. to stretch.

Temporary Activity – Interaction

This dimension captures the level of involvement of the user in the temporary activity. The activity can be *passive*: does not require the user to perform any action, e.g. watching an animation. Or, it can be *active* and require users to interact with the computer, e.g. tagging a photo, browsing the web or playing games.

Temporary Activity – Determination of completion

The temporary activity may be *finite* or *not*. Finite activities have an identifiable end, such as tagging a photo, updating one's status. Non-finite activities do not have an explicit end, e.g. web browsing or reading a newspaper. They may be best accomplished during an undetermined length of time. Finite tasks may allow smoother return to the primary activity.

Participatory Design Sessions

We conducted five participatory design sessions to better understand the experiences of users when they wait. We selected the groups (22 users total) through

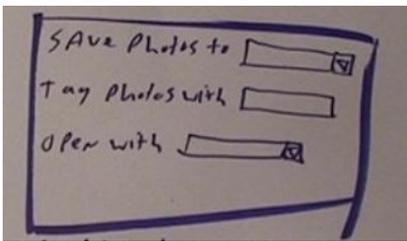
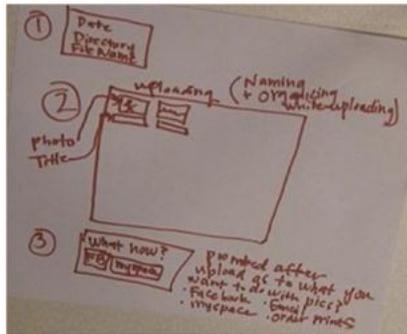


Figure 2: Paper prototypes for launching secondary activities while downloading photos



Figure 3: Paper prototype for booking a flight with contextual information during wait periods.

an iterative process. We started with two groups of computer scientists, based on the results of the preliminary questionnaire and the two first participatory design sessions, we reworked the taxonomy and identified professional groups more likely to shed light on the design space. Specifically, we had heard that downloading and editing pictures were common activities that required waiting; therefore we selected a group of photographers. We had also heard that people were interested in managing their time during the wait, so we worked with a group of managers. After starting with programmers we focused on groups with less extensive technological training.

In each session, users were asked to identify and describe three typical waiting situations before brainstorming on activities they may want to conduct to improve how they spend their waiting time. The brainstorming was situated in up to five predefined user cases, and their own identified situations.

The most common waiting situations reported by participants were *loading software, installing/updating a program, loading webpage, downloading pictures from a camera, and booting a computer* (Figure 2). In general, users reported a great diversity of situations, such as *ripping and burning CDs, connecting to Wifi, syncing a phone, streaming web content, doing a backup, booking a flight, rendering video and logging off*.

Participants in different groups mentioned the importance of *avoiding waiting time* as much as possible. This point was particularly stressed by the group of managers. The solutions proposed by this group included to schedule the booting up of their computer a few minutes before their arrival and scheduling soft-

ware updates after their departure from work, as well as to rank the applications to be launched at startup to get working immediately.

Several participants suggested temporary activities that were in the *same context* as the primary activity. However, to avoid interfering with the primary activity, these temporary ones should remain *passive* (not require any action). One participant proposed the situation of booking a flight ticket, which led to a large number of ideas. Most consisted of providing contextual information about the travel destination during the wait time, such as top tourists activities, weather, local headlines and events in the area (Figure 3).

In two different groups, several participants mentioned the fact that in many cases, they did not care about staying in context or in the flow. Indeed, they commented that these waiting periods could serve as breaks since they already used them as such when they knew the wait would be long. They explained that they could use their machine's booting up time to get coffee and greet colleagues, or a large software's loading time to browse the internet.

One of the predetermined user cases was the inclusion of some type of entertainment application in the progress bar. Overall very few participants responded favorably to this suggestion. The entertainment ideas mentioned were to show feeds from Facebook, news headlines, RSS feeds or the word of the day.

Survey

We surveyed 96 participants (47 females) on broader user preferences about temporary activities in different situations. Situations came from the discussions with

our participants during the workshops. Most of them divided their activities in “at work” or “at home”. While the terminology used may not be adequate (“at home” can be seen as the lunch time at work premises or “at work” can be seen as a working day out of the office), most of our participants use this simple classification to characterize their secondary activities and think about them.

47% of the participants mentioned staying idle for short waiting period (<5s), while 37% of the participants reported switching to a temporary activity. However, 65% answered that they switch to other activities for longer wait (> 15 s). Surprisingly only 50% preferred to do something else when the waiting period was more than 1 minute.

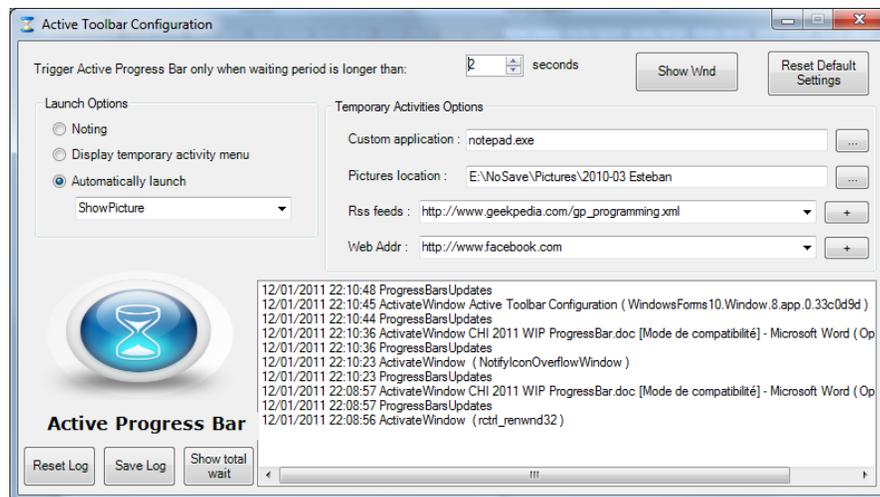


Figure 4: The Configuration window allows users to set their preferences about when to trigger the progress bar, what activity to trigger, and various parameters for those temporary activities such as which directory to use for the photo slideshow. A debugging window shows an event log.

Active Progress Bar Design

65% of our participants indicated a willingness to use an active progress bar to help them switch to a temporary activity. They estimated that it was worth using the tool when the wait was over 60 seconds. 66% of the participants answered that the temporary activity should be customizable as a function of the application that makes you wait. 70% of the participants answered a preference for switch back automatically to their primary activity at the end of the waiting period. 36% of the participants indicated a desire to switch back to their activity on their own, by retrieving the appropriate window. However, they mentioned that they wanted to be notified that the wait was over. Most of the participants preferred a visual artifact that displayed the end of the waiting period.

Desired Activities when Waiting

In work settings, 60% of our participants answered that they would prefer to stay in context and perform a work related activity or manage their to-do list and calendar during a wait of more than 1m. During a medium waiting period of 15s, 40% of the participants would rather perform a passive activity such as getting information about the program waiting or about the weather or traffic. Only 15% of our participants mentioned that they would perform similar passive activities for short waiting periods. At home, 67% of our participants favored activities such as reading twitter or news feeds, in particular for longer waiting periods.

Active Toolbar Prototype

We developed an active progress bar prototype to investigate some of the ideas generated by participants. This software monitors the operating system (OS) and when it detects a waiting period proposes temporary

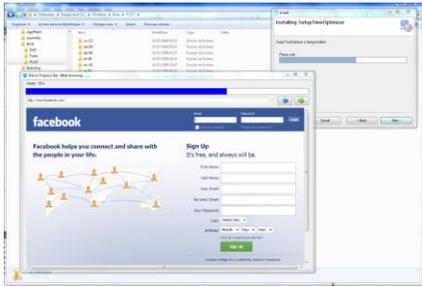


Figure 5: The user has configured the Active Toolbar to popup a browser window with Facebook. While waiting for a program to install (top right), he can login. Next time he needs to wait, he will see updates from his friends.



Figure 6: The screenshot displays a large directory was being copied. At the bottom left appears the traditional progress bar produced by Windows Explorer. Active Toolbar window shows a slide show of photos from a favorite directory. Users can remain passive and enjoy the pictures for a few seconds, or be more active and rename files, a common way to annotate photos.

activities that the user can customize (Figure 4). Since a common way to notify the user of a waiting period is to display a progress bar control that fills over time, our prototype monitors and evaluates all the system progress bars lasting longer than a second in the OS. Users can select the value in seconds of the minimum estimated waiting time that should trigger the active progress bar. User can choose if they want the temporary activity to start automatically when the wait is detected or be presented with a menu of activities.

Our Active Toolbar prototype proposes samples of secondary activity applications, combining existing applications with a normal progress bar. Entertainment activities include games such as Sudoku, a web browser with Facebook (Figure 5), and a photo viewer (Figure 6), while work activities include FreeWriter, which allows the user to manage a to-do list, or an RSS feed. Users can also choose any other application of their choice to be launched, but it would not include the integrated progress bar indicating the remaining time.

Conclusion

In this paper we introduced active progress bars, a new approach facilitating the switching to temporary activities when a primary activity requires users to wait. Our contributions are:

- a multifaceted study providing evidences that a subset of users see a benefit in switching to a temporary task while waiting and welcome the idea of getting assistance in switching tasks;
- we provided a taxonomy of waiting times;
- we explored the design space of an active progress bar tool using four approaches (participatory design, survey, informal discussions, and prototype);

- we demonstrated the feasibility of detecting the waiting with an active progress bar prototype.

In summary, we believe that with adequate user control, active progress bars have the potential to help users make better use of their time while waiting. We look forward to further development of the prototype and hearing users' feedback following its deployment.

REFERENCES

- [1] Csikszentmihalyi, Mihaly. Flow: The Psychology of Optimal Experience. HarperCollins, 1991.
- [2] Czerwinski, Mary, Eric Horvitz, and Susan Wilhite. "A diary study of task switching and interruptions" In Proc. of CHI '04: Proceedings of the SIGCHI conference on Human factors in computing systems (2004) 175-182.7
- [3] Galletta, D., Henry, R. M., McCoy, S., Polak, P., Understanding the direct and interaction effects of web delay and related factors, in Zhang, P. Galletta, D. F. (Eds), Human-Computer Interaction and Management Information Systems: Foundations (Advances in Management Information Systems), M. E. Sharpe, Armonk, New York (2006), 5, 29-69.
- [4] Harrison, C., Amento, B., Kuznetsov, S., and Bell, R. 2007. Rethinking the progress bar. Proc. of the 20th Annual ACM Symposium on User interface Software and Technology (2007) 115-118.
- [5] Harrison, C., Yeo, Z., and Hudson, S. E. 2010. Faster progress bars: manipulating perceived duration with visual augmentations. Proc. of the 28th international Conference on Human Factors in Computing Systems (2010) 1545-1548.
- [6] Jing Jin and Laura A. Dabbish. 2009. Self-interruption on the computer: a typology of discretionary task interleaving. In Proceedings of the 27th international conference on Human factors in computing systems (CHI '09). ACM, New York, NY, USA, 1799-1808.