Textured Agreements: Re-envisioning Electronic Consent

Matthew Kay, Michael Terry
Human-Computer Interaction Lab
University of Waterloo
Waterloo, Ontario, Canada
{mjskay, mterry}@cs.uwaterloo.ca

ABSTRACT

Research indicates that less than 2% of the population reads license agreements during software installation [12]. To address this problem, we developed *textured agreements*, visually redesigned agreements that employ *factoids, vignettes*, and *iconic symbols* to accentuate information and highlight its personal relevance. Notably, textured agreements accomplish these goals without requiring modification of the underlying text. A between-subjects experimental study with 84 subjects indicates these agreements can significantly increase reading times. In our study, subjects spent approximately 37 seconds on agreement screens with textured agreements, compared to 7 seconds in the plain text control condition. A follow-up study examined retention of agreement content, finding that median scores on a comprehension quiz increased by 4 out of 16 points for textured agreements. These results provide convincing evidence of the potential for textured agreements to positively impact software agreement processes.

Categories and Subject Descriptors

H5.m. [**Information interfaces and presentation (e.g., HCI)**]: Miscellaneous. K.4.1 [**Computers and Society**]: Public Policy Issues – *privacy.*

General Terms

Design, Experimentation, Security, Human Factors, Legal Aspects.

Keywords

Informed consent, end-user license agreement, EULA.

1. INTRODUCTION

Less than 2% of the population reads the end-user license agreements (EULAs) commonly shown during software installation [12]. However, these agreements often contain terms of direct concern to users. For example, agreements may describe data collection policies, or inform the user that additional software will be installed, such as digital rights management (DRM) software or software

Symposium On Usable Privacy and Security (SOUPS) 2010, July 14-16, 2010, Redmond, WA, USA.

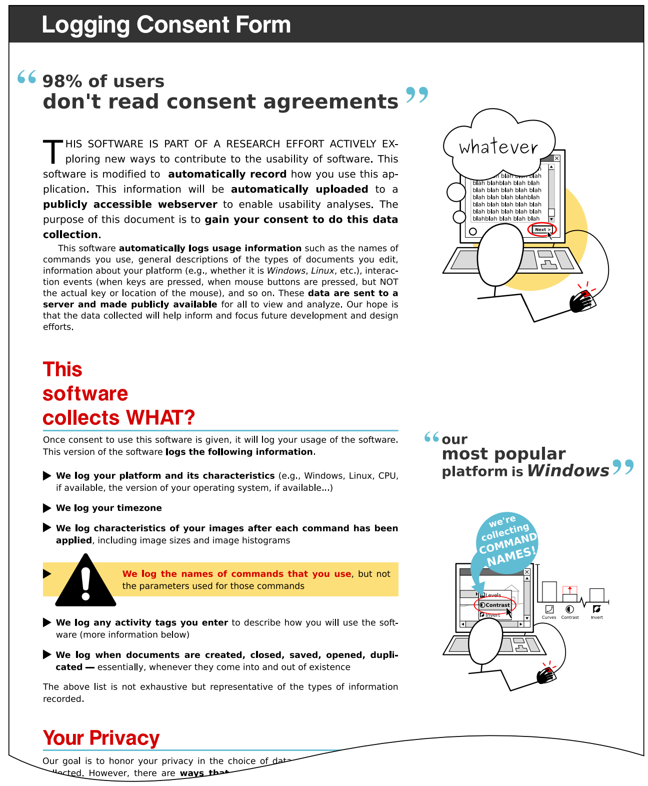


Figure 1. Excerpt from an example textured agreement. Basic typographic manipulations are applied: headings, bullets, bold text, and a more distinct lead paragraph. Important content is also made more salient using factoids, vignettes, and iconic symbols.

that delivers targeted advertising. Although these agreement terms can subsequently have an impact on the user's privacy, security, or overall user experience, the terms are typically not read, as they do not contribute to the user's more immediate goal of using the software to accomplish a specific task [8,9].

While some companies have little desire for end users to fully understand the implications of their software agreements, a growing number of proprietary software companies are presenting agreements in ways that are more easily understood by their users. For example, the makers of Aviary (a web-based suite of multimedia applications) provide a license agreement that includes a summary of the agreement's terms in a side column [3]. This represents a concerted effort to build a trusted relationship between the company and its users through an improved software agreement process.

Besides producers of proprietary software, there are a wide range of other software producers who would benefit from and welcome an improved software agreement process. For example, producers of free/open source software (such as the Free Software Foundation[1]) often wish others to learn about their movement and its philosophical underpinnings. Methods for better communicating the terms and perceived benefits of their licenses—such as the GNU General Public License [10]—would help further their cause.

As a second example, researchers who conduct studies using remotely distributed software (such as the ingimp project [22]) have an ethical obligation to obtain informed consent from their participants. However, if participants ignore the consent agreement included with the software, they may not fully understand the risks associated with participating in the research; this can include the potential for sensitive information to be inadvertently made public [22]. There is a clear need for an improved software agreement process in these situations.

This paper presents work that helps address these problems through visually redesigned software agreements that we call *textured agreements*. Without modifying the original content, textured agreements employ visual design techniques such as typography and layout to create a well-defined information hierarchy. A set of visual design strategies are also employed to help capture and retain reader attention. For example, textured agreements use warning symbols to highlight terms of an agreement that may affect the user's privacy, and employ visual variety to create interest throughout the document (Figure 1).

Textured agreements arose out of an iterative design process informed by formative evaluations. To validate the resultant designs, we conducted two between-subjects laboratory experiments. Both experiments employed deception via a distractor task that included the need to install software and agree to its terms.

In our studies, we found that reading time was increased by 30 seconds in the textured agreements compared to a plain-text agreement. We also found an improvement in comprehension of agreement terms. Finally, we found that summaries of agreements were effective at capturing attention and communicating information, but at the cost of reading the full agreement. In essence, summaries minimize the perceived need to read the full agreement.

In the rest of this paper, we first review past work in improving software agreements, then describe the design methodology that led us to create textured agreements. We present the specific techniques that compose textured agreements, then describe the two experimental studies conducted to evaluate the effectiveness of the agreements. Results and implications from both studies are presented, along with perspectives from our local internal review board on the feasibility of deploying textured agreements in real-world situations.

2. BACKGROUND

Software agreements are employed in a range of situations, including website privacy policies [6,16,17,19], EULAs shown during software installation [12,11,13], and consent agreements accompanying research software [22]. Recognizing that there exists a wide range of such agreements (privacy policies, EULAs, consent agreements), we collectively refer to these types of agreements as *software agreements*.

As previously mentioned, there are numerous motivations for improving software agreement processes, including economic, ideological, and ethical motivations. Governments are also increasingly considering legislation to both standardize and improve software agreements to help protect consumers [20]. This presents an additional motivation for research in this space: if legislative measures are desired, they should be crafted in a way that results in demonstrably better software agreement processes.

Recognizing the inadequacy of current software agreement methods, a number of strategies have been developed in both industry and the research community to improve this process. Many of these strategies are aimed at reducing the demands associated with reading lengthy agreements, as we describe next.

To assist users in comprehending the terms of lengthy agreements, Good *et al.* drew inspiration from techniques used in fields such as medical informed consent [15] and financial privacy notices [1] to create single-screen *summaries*[2] of EULAs [12,11]. These summaries have been found to significantly reduce the number of installations of spyware when presented as part of the software installation process. However, as we will show later, this approach may run the risk of reducing the likelihood that users read the full agreement should they proceed past the summary.

Kelley *et al.* [17] took inspiration from nutrition labels in designing a summary of privacy information for P3P (Platform for Privacy Preferences)-compliant websites. These "privacy labels" use a table of privacy policy information to indicate how users' data is used by a website and what data collection practices users can opt in or out of. While these labels were found to improve users' ability to find relevant information in web privacy policies, it is unclear how well the technique would generalize to other forms of software agreements.

Patrick and Kenny approached the problem of lengthy agreements by proposing "Just-In-time Click-Through Agreements" (JITCTAs) [20]. Instead of requiring the user to agree to a large agreement prior to using the software, smaller JITCTAs are displayed when performing an action that causes personal information to be collected by the software. In essence, this approach segments the agreement into smaller components that are shown at situationally appropriate times, increasing the chance that users will notice, comprehend, and act on the agreement terms. While promising, this technique is only appropriate when an agreement can be broken into smaller context-sensitive pieces that do not overload the user with requests as they work.

To address the problem of capturing user attention in warning dialogs, Brustoloni *et al.* [4] created *polymorphic dialogs*, which rearrange the order of buttons on the interface or temporarily make buttons inactive. These strategies are intended to prevent users from learning a fixed path through the interface. This approach has been shown to increase the likelihood that users take the time

1 http://www.fsf.org/

2 The summary approach is often called "layered notices" [11,12,15,19,21], with the implication that it could be extended beyond simply presenting a summary with the full text, to presenting multiple, progressively more detailed layers—an approach that has not, to our knowledge, been tested.

to understand dialogs that outline the security risks corresponding to the user's intended action (for example, opening an email attachment that may contain viruses). However, it is unclear whether similar techniques would be effective in the context of software agreements, where the content is longer and denser.

In contexts outside of software agreements, research has explored a number of other approaches aimed at helping individuals understand lengthy agreements. To understand how to improve comprehension of medical agreements, Campbell *et al.* [5] compared the efficacy of video, text, and enhanced text agreements that used improved headers, pictures, and other visual enhancements. They found that subjects with lower literacy rates were able to better comprehend the enhanced text agreements than the other treatments. Interestingly, video performed worse than standard text-based agreements. However, it is unknown if these results would transfer to the context of software agreements, particularly since participants in this study were explicitly asked to read the agreements; a significant problem with the software agreement process is simply compelling users to read the agreement.

While this past work has made some important strides to improving the software agreement process, there is still much room for improvement. For example, while summaries have been shown to be effective in some circumstances, summaries do not, by definition, include all terms of an agreement. Thus, in cases where users wish to proceed with software installation, they are still faced with a lengthy agreement. The visually enhanced medical agreements of Campbell *et al.* show promise in addressing this problem, but these have not been tested in software contexts.

Among the many approaches that have been explored, visually enhanced agreements represent a particularly attractive avenue for further research: current and proposed solutions to the software agreement process are largely conducted using the medium of *plain text*, with minimal use of typography or other visual design techniques applied directly to agreement content. Given the rich history of visual design, there is an opportunity to draw on this field to create software agreements that users find more compelling and easier to read. We turn now to the research we conducted to explore this particular approach.

3. DESIGN PROCESS

To explore the space of possibilities for improving the software agreement process, we started by engaging in rapid, low-fidelity prototyping and formative evaluations of a range of alternative methods for presenting agreements. 21 subjects participated in this phase of the research, evaluating dozens of prototypes employing techniques ranging from supplementary videos and illustrations to enhancements of the text-based agreement itself. We explored this design space by drawing inspiration from related fields (e.g., advertising, technical communications, and comics) and the various approaches attempted in previous work (e.g., summaries and video).

We used paper-based prototypes and computer-based mock-ups to explore the potential effectiveness of different approaches. These prototypes were presented as part of a mock software installation process, mimicking the standard context in which users encounter software EULAs. Participants in the formative evaluations were asked to imagine they were downloading and installing the software (computer-based prototypes actually included simulated installers), for the purpose of evaluating the software's usability. That is, participants were not initially informed that our focus was on understanding the effectiveness of the altered agreements in capturing and retaining users' attention. Participants typically viewed multiple prototypes, so this minor deception was effective for only the first prototype; accordingly, each subject was shown the prototypes in a different order. This same basic approach of supplying a distractor task was utilized in later experiments to attempt to replicate realistic scenarios of encountering software agreements.

3.1 Design Insights

Among the many prototypes tested, the visually redesigned text-based agreements appeared to be the most promising. When subjects were shown additional material, such as videos or separate illustrations depicting agreement contents, they either expressed confusion or disinterest in the additional content. In contrast, when shown text-based agreements that made use of typography and graphic design, subjects often stated they felt the software distributors were *making an explicit effort to communicate with them*. Subjects saw these enhancements as an attempt to make the agreement process accessible and meaningful to them.

As a result of these responses, we decided to focus our prototyping efforts exclusively on cultivating a set of visual design strategies tuned to improving the software agreement process. In order to do so, we drew inspiration from visual design techniques used in popular media, such as magazines, newspapers, technical manuals, and how-to guides. These media face similar challenges in enticing their audiences to take notice and retaining that attention once captured.

A rich repertoire of visual design practices has been developed to suggest the value in taking notice of information and to improve the accessibility and readability of informationally dense documents. For example, Tufte advocates the use of "separation" and "layering" of information when visualizing dense data sets graphically [23]; these concepts refer to constructing visualizations with visually distinct layers that are both coherent as a whole and can be read separately, where the usefulness of data on lower layers is not lost in higher layers of the visualization. In a similar vein, Kress and van Leeuwen discuss "salience" and "framing" as ways to increase the visual weight of data that is higher in the "hierarchy of information," making it possible to read a composition at a higher level [18]. They also stress the importance of composition in facilitating non-linear reading of dense documents, which allows readers to more quickly find personally relevant information [18]. In a survey of research into the effectiveness of various methods of presenting technical information, Wright similarly found that headings and other design elements can make skimming a document easier, and even assist in the integration of content for readers who do not skim [26]. Wright also notes that design can be used effectively to influence the pace at which a dense document is read [26].

We will expand upon the use of these and other techniques in the next section, wherein we present our approach to visually redesigning software agreements.

4. TEXTURED AGREEMENTS

In this section, we introduce *textured agreements*, visually enhanced software agreements. We divide this section into two subsections:

1) A description of the overall goals we wish to achieve with the visual redesign of software agreements, and 2) a description of the specific techniques employed to reach these goals.

4.1 High-Level Goals

In redesigning software agreements, we wish to accomplish two high-level goals. First, we wish to introduce an *information hierarchy* to impart a clear visual organization to the material. This information hierarchy should serve to highlight important information in the agreement and suggest its personal relevance to the reader. Second, we wish to *capture and retain reader interest*. Our chief means of accomplishing this latter goal is through the use of *visual variety* in the designs.

4.1.1 Provide a Clear Information Hierarchy

Given the typical information density of a software agreement, there is a need to provide a clear visual hierarchy to its contents. For our purposes, we will broadly consider the concept of a “visual information hierarchy” to encompass the principles behind Tufte’s “layering” and “separation” techniques in constructing visualizations [23], and Kress and van Leeuwen’s use of “salience” and “framing” [18] in visual design.

A good information hierarchy will lend agreement terms of greater importance greater visual weight, where importance is defined from the user’s perspective. For example, terms that describe features that affect the user’s privacy, such as the automatic collection of personal information, should be emphasized over more common clauses that users will expect to be in an agreement, such as standard limitations of liability.

Imparting a clear and distinct visual hierarchy helps counter common problems with existing text-based agreements: the impression that documents are irrelevant, the inability to skim their content with ease, and the impression that they are long, difficult, and cumbersome to read. When done well, the resultant visual hierarchy yields a clear path for navigating the content, making it easier to locate pertinent information and giving an impression that the document contains information that is interesting and worthwhile. The full range of techniques we employ to create this visual hierarchy—including typographic manipulation, white space, factoids, vignettes, and iconic symbols (Figures 2-4)—are described in detail later; we provide a brief illustrative example here.

It is difficult for users to skim a plain-text agreement to find information they consider personally relevant as everything is contained in long paragraphs. To break up these swaths of text and highlight information likely to be of interest to users, we can use “factoids”, or tidbits of interesting information that stand apart from the text. For example, consider an agreement that describes the data collection policies of some software, where one piece of data collected is the user’s operating system. We can highlight this fact with a factoid that states the most popular platform amongst the software’s users (Figure 2). This introduces an interesting and memorable bit of information into the agreement and simultaneously clues the user into a specific consequence of accepting the agreement, lending a sense of transparency.

Note that this factoid is also positioned directly next to the agreement text that describes the data logged by the application, illustrating an important feature of a good information hierarchy: readers

Figure 2. Factoids present interesting information related to the nearby content in the consent agreement.

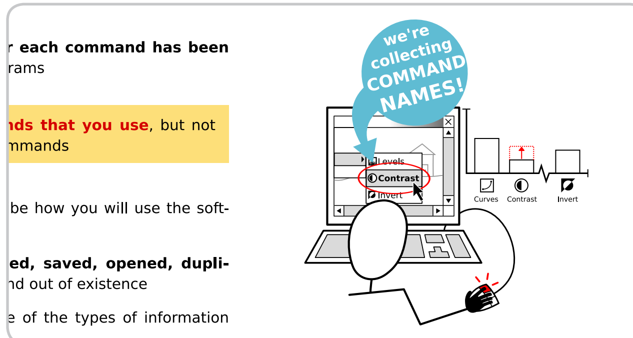


Figure 3. Vignettes draw the reader in and communicate agreement content through mini-narratives.

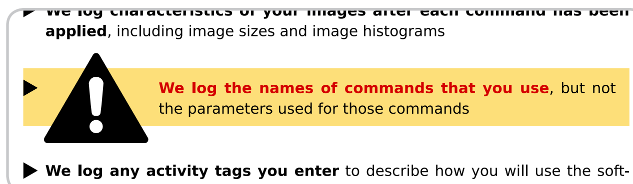


Figure 4. Iconic symbols (e.g. warnings) highlight important information.

can readily transition between levels of the hierarchy. More specifically, readers can enter the document at a high level (say, by skimming headers and factoids), then easily find related information nearby in lower levels. This “skimmability” lowers the barrier of entry for the entire document: when readers feel they can skim the agreement without negotiating paragraphs of text—and actually find interesting information while skimming—there is a greater chance they will subsequently read further into the content.

4.1.2 Capture and Retain Attention with Visual Variety

While the need for an information hierarchy dictates that we should raise information to higher levels of the visual presentation, as we begin to raise more elements higher in the hierarchy, we risk overusing our techniques. For example, during formative evaluations, we found that users would not read more than 3 or 4 warning boxes (Figure 4) per document; overuse of those elements rendered them unable to retain interest. It is therefore important to employ *visual variety* when selecting amongst techniques in order to continually *capture and retain interest* in an otherwise lengthy document. For example, one could use headlines, warning symbols, or vignettes to highlight privacy information in an agreement. The choice of which technique to use, where, and when, is partially informed by the need to create visual variety in the overall design.

At a large scale, textured agreements use *progressive exposure to different visual design techniques* to create visual variety. New visual elements are continually introduced throughout the agreements

“98% of users
don't read consent agreements”

Figure 5. An example of a factoid used at the top of the textured agreement in Figure 1 to draw readers in.

to entice readers to take notice. Devices like factoids and vignettes (Figure 3) are particularly effective means for introducing variety, since they are independent of the primary body of text and can thus be placed nearly anywhere (within reasonable proximity to related information). Balancing the use of these various techniques is important: concentrating all instances of a technique in one area of the document will render it ineffective. The need to maintain this balance constrains how much information can reasonably be placed at higher levels of the document.

When a document uses an information hierarchy and visual variety together effectively, it suggests an accessible *pacing*—or reading rate—for that document. Pacing influences a reader's assessment of the effort required to read a document. As an example, the frequency of headings can communicate to the reader how quickly a document can be read, and how easily it can be skimmed [26]. Plain-text agreements—with no headings, or typographic manipulation in general—create an impression that the document is lengthy and arduous to read [26]. Through application of the aforementioned techniques, textured agreements create a *textual pattern* that suggests one can easily move through the document, at various levels of detail, to glean the most relevant information from it.

Having established these high level goals, we now describe how to select and apply the various techniques to satisfy these goals.

4.2 Specific Techniques

The primary techniques used to achieve the goals described above are typographic manipulation, pull-quotes, vignettes, and iconic symbols (Figures 2-4). While these methods are commonly used in other media (see the aforementioned [18,23,26]; practical examples of various techniques are also discussed elsewhere [7, 24]), our contributions lie in the selection and adaptation of these strategies to the design of software agreements, and in the evaluation of the effectiveness of these techniques. In this section, we describe how these techniques are used to create textured agreements[1].

4.2.1 Typographic Manipulation

Given that software agreements are most commonly presented as plain text, it is not surprising that the overwhelming impression among users is that agreements are both irrelevant and boring. As a first step towards improving the presentation of an agreement, we can break up unformatted and uninviting "walls" of text using simple typographic manipulation: we can bold key terms and phrases, and add headers and bullet points. For example, the body text of the agreement in Figure 1 has been made more inviting and relevant by adding headers and bolding key terms of interest to readers. These additions create the first level of the information hierarchy: readers can now skim key phrases and headers in the document.

1 Further examples of these specific techniques, along with template agreements to facilitate adoption, are available on our website: http://hci.uwaterloo.ca/research/textured.

Figure 6. An example of a vignette used to similar effect as the factoid in Figure 5.

4.2.2 Pull-Quotes and Factoids

While typographic manipulation has made this agreement more inviting, we would like certain information to be even more accessible to users when skimming the agreement. *Pull-quotes* are catchy or interesting quotes taken from the primary text, shown separately from the main text. Pull-quotes use a different (often larger) font to make them visually distinct and noticeable. They are used in a variety of media, such as websites, newspapers, and magazines, to improve the skimmability of documents [24] and to convey key concepts through quickly consumed, "bursty" nuggets of information. In the case of software agreements, the text is typically not pithy (and in fact is generally rather dry), making actual pull-quotes impractical. Instead, we take the approach of writing *factoids,* which are laid out similarly to pull-quotes, but do not use text taken directly from the agreement body.

In our agreements, we have taken two approaches to writing factoids: highlighting interesting facts that arise as a consequence of agreement terms—such as the earlier example where we hint at data collection by stating the most popular platform among the software's users (Figure 2)—and using humorous facts to hook users at the top of the agreement, such as a factoid exclaims, "98% of users don't read consent agreements" (Figure 5). In our evaluation we will see that the latter factoid contributed to many users reading the rest of the document: its use of humour simultaneously acknowledges the tedium of reading a software agreement while suggesting that this particular agreement is different, accessible, and worth reading.

It is important to keep factoids germane to the software agreement or the agreement process. At the same time, factoid content should be chosen and worded to suggest that the document is *accessible* and includes interesting and relevant information relevant to the user, rather than just boilerplate legal text. When done effectively, factoids will lend a human, personal element to the agreement, elevate important agreement content to a higher level of the hierarchy, and pique reader interest in the agreement content.

4.2.3 *Vignettes*

Vignettes are mini-narratives related to the content of the agreement. As with factoids and pull-quotes, vignettes help both to capture user interest and to create a clear information hierarchy by raising important content to higher levels of the hierarchy. One of the potential strengths of vignettes is to make a more direct, literal connection with the reader, since vignettes depict users interacting with the software (or even with the software agreement) through a comic-like medium. For example, the agreement in Figure 1 employs a vignette that depicts the user clicking on the contrast tool. In the vignette, we highlight that this information is collected with the exclamation, “we’re collecting COMMAND NAMES!” (close-up in Figure 3). The informal nature of the illustration, its suggestion of an underlying narrative, and its sensational text add interest, while the involvement of the reader by implication (as a potential user of the software) adds personal relevance to the content.

4.2.4 *Iconic Symbols*

Operational manuals and technical manuals often use warnings to alert readers to information vital to their personal safety. Warnings are similarly used in the workplace (see Wogalter and Laughery’s summarization of research on workplace signs and labels for more on the topic [25]). These warnings are frequently accompanied with an icon to demarcate and classify the warnings (as in Figure 4).

To help capture reader interest and create an information hierarchy, textured agreements use warning symbols and colored boxes to highlight particularly sensitive information in the software agreement, with the hope that even those who quickly skim the document will stop and read the content associated with the warning.

Our formative evaluations suggest that warning symbols must be used sparingly to ensure they are perceived as highlighting truly exceptional or “hazardous” features of the agreement. Otherwise, users become desensitized to them and read only the first few warnings they encounter. This is consistent with previous work showing that warning symbols must be used only with truly hazardous information in order to retain their “arousal strength” (e.g., [2,14,25]).

4.3 Summary

The techniques discussed above must be applied with the overarching goals of building an information hierarchy and balancing visual variety. When this is done well, it is possible to break down the monotonous presentation typically associated with software agreements, creating a more interesting and accessible document. Note that there is no single template to achieve these goals. Instead, textured agreements represent a set of *strategies* one applies to the visual design of a software agreement to achieve these ends.

In the sections that follow, we present results from two experiments which suggest the effectiveness of these techniques.

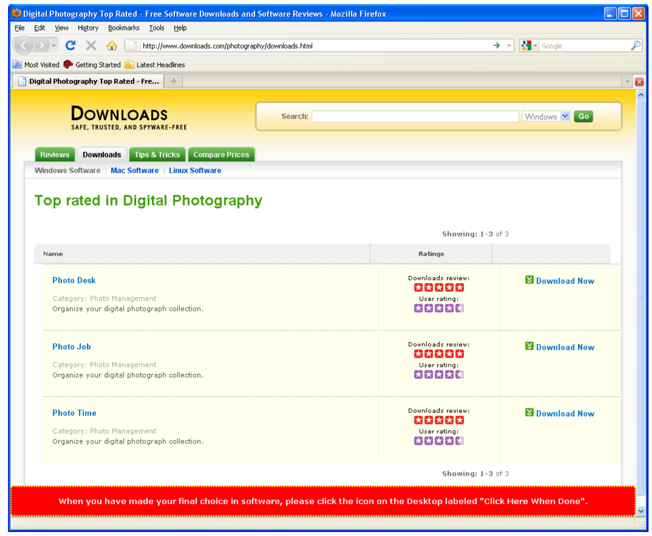


Figure 8. Mock web page shown to participants in experiment 1. The order of the application names on this website was counterbalanced between participants.

5. EXPERIMENT 1

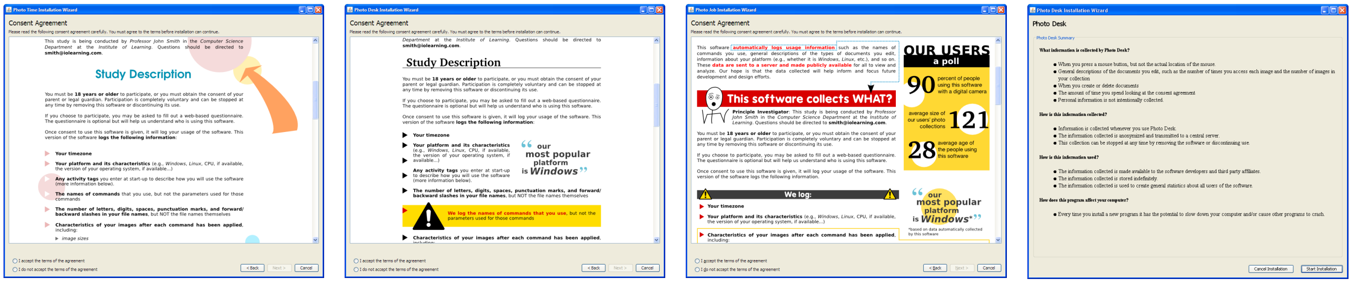
In this first experiment, we were primarily interested in understanding textured agreements’ overall effectiveness in capturing and retaining readers’ attention compared to existing approaches.

5.1 Experimental Design

A between-subjects deception experiment was devised to evaluate the ability of textured agreements to capture reader attention. The study employed five conditions corresponding to five different agreement styles: three conditions with textured agreements’ visual techniques applied to varying degrees (minimal, moderate, heavy), a pre-installation summary condition similar to that of Good *et al.*’s study [12], and a control condition with plain-text software agreements. The summary condition was included to partially replicate Good *et al.*’s previous study and provide another point of comparison for the textured agreements.

Subjects were asked to download, install, and use three image manipulation applications from a mock web page (Figure 8) for the purpose of choosing the best application for rotating images in a digital photo collection. Each application’s installer was a custom-written installation program that could be experimentally manipulated to show a different agreement. The installer was instrumented to collect interaction events—in particular, the time spent on each screen of the installer and scroll events in the agreement itself. After choosing an application, participants completed a questionnaire.

To increase ecological validity, we carefully designed the study to minimize the chance that participants would artificially focus on the software agreements. We took two measures to avoid this potential bias. First, deception was employed. Subjects were told that the purpose of the experiment was to learn how they choose software when multiple choices exist. Second, a verbal consent script was used to obtain initial consent to participate in the study. This was motivated by the observation during our formative evaluations that written consent agreements primed users to look at the software agreements: reading and agreeing to a paper consent agreement made them more cognizant of text-based consent processes



A. Minimal B. Moderate C. Heavy D. Summary

Figure 7. Examples of each experimental condition from experiment 1. For the textured consent agreements (A-C), approximately the same portion of each agreement is shown. For the summary condition (D), the initial summary screen is shown.

in general. Subjects were debriefed upon completion about the true nature of the experiment and given a second, paper-based consent form to provide consent to keep their data.

5.2 Procedure

Subjects were given a written scenario and instructions after obtaining verbal consent. The scenario indicated that they had recently received a digital camera, but lacked software to perform basic manipulations of the images. Accordingly, they were told to imagine they had just found a website with three different photo applications. The instructions indicated that they could download, install, and use any of the applications. Their specific task was to decide which software they would choose to use for the purpose of rotating images. A folder of improperly-oriented photographs was provided to assist with their evaluations. Once they had made a choice, the instructions indicated that they should start a questionnaire to record their final selection.

After receiving and reading the instructions, subjects had the opportunity to ask questions. They were then seated at a desktop computer with a web browser already opened to the download web page to perform the task. After completing the task and the subsequent questionnaire, they were debriefed about the actual intent of the study and given a written consent agreement.

5.3 Experimental Conditions

We developed three separate plain-text software agreements for the applications, drawing from existing software agreements. The content was designed to be consistent in form and presentation across all three agreements. Each agreement indicated that the application was instrumented to collect data, though the specifics of what was collected, why, and by whom, varied per agreement.

Three textured agreement *templates* were developed representing minimal, moderate, and heavy application of the techniques. These templates were applied to each of the plain-text agreements, yielding three separate instantiations of a template per condition (Figure 7 gives examples of each experimental condition).

In more detail, the conditions in the experiment were as follows:

1. *Control Condition*. A plain-text software agreement.
2. *Minimal Condition*. Based on lessons learned in the formative evaluation, we hypothesized that merely adding visual decoration to a software agreement would not be enough to increase reading times. Instead, our experiences suggested that one needed to conscientiously apply the textured agreements' techniques. To test this hypothesis, the minimal condition represents an aesthetically pleasing software agreement with visual adornments. However, while aesthetically pleasing, the visual design does not otherwise strive to reinforce the *content* of the agreement. (Figure 7A).
3. *Moderate Condition*. This condition represents what we feel is a balanced application of the strategies of textured agreements. This agreement uses a more scholarly heading font, warning boxes for three important agreement clauses, and factoids relevant to agreement content. (Figure 7B).
4. *Heavy Condition*. This condition was designed to incorporate as many of the techniques as possible to make a visually dense, deeply layered agreement (Figure 7C).
5. *Summary Condition*. One-page summaries for the three software agreements were developed using the heuristics used by Good *et al.* [12,11] (Figure 7D).

To avoid order effects, the order of agreements paired to installers was counterbalanced. The order of the software applications' names on the web page was also counterbalanced. To avoid potential effects due to pairing the same application names to the same agreements, there were six different application name-agreement pairings (applications were given the fake names "Photo Time", "Photo Job", and "Photo Desk" to avoid the effect of brand recognition; consequently, we had to vary application name-agreement pairings to avoid effects caused by one name bring more intrinsically attractive than another). This yielded a 5x6 between-subjects design with 5 conditions and 6 application name-agreement orderings.

5.4 Performance Measures

We measure the effect of each treatment by recording the amount of time subjects spent on individual installer screens and the maximum amount the software agreement was scrolled.

Because we strove to create a more realistic experience, users could run each installer multiple times. However, this capability complicates measures of time spent in the installer screens. Accordingly, our timing measures are derived from the *first* time an installer is run, which we refer to as *first-run timings*. This measure is likely to more closely correspond to actual practices, since people typically only run an installer once. For each subject, an average timing is calculated from first runs of the installers. If a subject does not run a particular installer, it is not included in the calculation of the subject's average time. The maximum amount an agreement is scrolled is calculated the same way.

5.5 Experimental Apparatus

The study was conducted using a basic Windows XP installation on a VMWare virtual machine that had the Internet connection disabled. Using the snapshot feature of VMWare, we were able to have identical start conditions for every subject. A setup script was run before each session to set the experimental conditions for each subject based on their subject number.

The applications' installer was a custom-developed installer written in Java. Its design used the Windows "look-and-feel", and mimicked the appearance and feature set of a typical Windows installer. The installer was instrumented to record when it was opened and closed, the time spent on each screen, interactions with controls, and whether subjects canceled or completed the installation.

With the exception of the summary condition, each installation process was identical. The first screen was a welcome screen; the second, the software agreement screen; the third, a screen informing the user where the software will be installed; the fourth, a screen showing the installation progress; and the fifth, a screen indicating the installation was finished. The summary condition added an additional screen, a summary of the software agreement, which was shown before the welcome screen. This ordering of screens mirrors that of Good *et al.*'s study, though we did not include a blank screen before the welcome screen in non-summary conditions. While Good *et al*. included this screen to act as a control for the presence of the summary screen, we wanted to increase the ecological validity of our study, and thus did not include it.

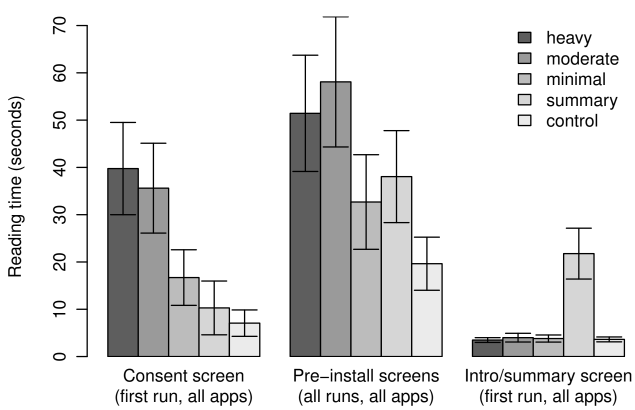


Figure 9. Mean time spent on installer screens (seconds with standard error): time on consent screen only; time on summary, welcome, and consent screens; and time on summary and welcome screens only.

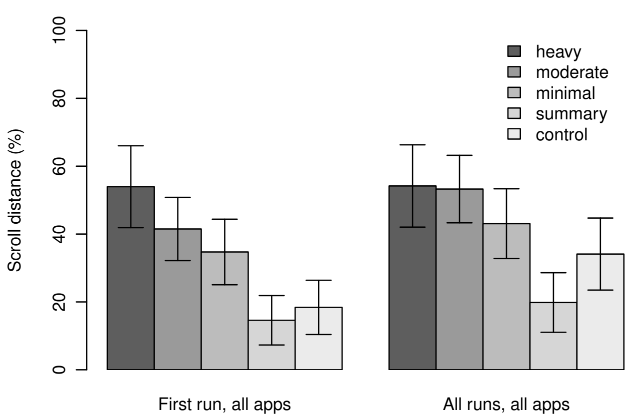


Figure 10. Maximum scroll distance into the consent agreement (% of document with standard error): average over first runs only and average over all runs.

The installation program did not actually install the application (the applications were already installed), but did copy all of the files to a temporary directory to simulate the installation process. It also added shortcuts to the Start Menu.

5.6 Participants

90 subjects were recruited in a university setting. Six dropped out, providing 84 subjects, or 17 subjects per condition, with the exception of the heavy condition, which had only 16. Subjects were compensated with a $10 gift certificate for a coffee chain. 43 females and 41 males participated, aged 17-47 years old (mean=24, SD=6). Subjects' self-reported computer expertise on a five point scale was an average of 3.4 (SD=1) with 5 being "most expert."

5.7 Results of Experiment 1

5.7.1 Timings, Scrolling Behavior, and Reading Self-Reports

An analysis of agreement screen timings indicates three outliers, one each in the minimal, summary, and control condition. These three subjects spent an average time of 250, 579, and 433 seconds, respectively, on the agreement screen, each of which is more than 3 standard deviations from the within-condition mean. These outliers were dropped from the timing analyses and are not represented in any graphs presented here. Apart from these outliers, we observed considerable variation in reading habits. In the questionnaire, subjects were asked to self-report their tendency to read software agreements in general on a five-point scale: "never noticed before", "never reads", "rarely reads", "often reads", or "always reads". We found this measure of reading habits to be a contributing factor to the scores, and thus include it as a factor in our analyses.

Figure 9 presents a plot of the first-run agreement screen timings for each condition. The heavy condition features the longest agreement screen time (mean=39.8 seconds, SD=39), followed by the moderate condition (mean=35.6, SD=39.2). Mean times in the summary and control conditions were 10.3 and 7.1 seconds, respectively. An ANOVA indicates significant differences between conditions ($F_{4,76}=5.65$, $p < 0.01$). Post-hoc Tukey tests indicate significant differences between: heavy and control ($p < 0.01$), heavy and summary ($p = 0.026$), moderate and control ($p = 0.026$), and a trend for significance between moderate and summary ($p = 0.063$). No other significant differences were found between conditions.

An ANOVA indicates significant differences between conditions for scrolling, as well ($F_{4,76}=3.96$, $p = 0.014$). Post-hoc Tukey tests indicate significant differences between heavy and control ($p = 0.04$), and heavy and summary ($p = 0.02$).

The questionnaire asked subjects to self-report how much they read each software agreement on a five-point scale. Self-reported reading amounts were found to be significant with respect to condition ($F_{4,76} = 3.16$, $p < 0.05$), with post-hoc analysis indicating the differences are due to subjects reporting they read the agreements more in the moderate condition than the control condition ($p < 0.05$).

5.7.2 *Aggregate Timings Across All Installations*

We noticed a tendency for subjects to cancel installations at the *summary* screen. This observation echoes Good *et al.*'s findings and suggests the effectiveness of the summaries in communicating information. However, this also slightly complicates comparisons of first-run time spent on the agreement screen, since subjects may not reach that screen the first time the installer is run. Accordingly, we also examined *total pre-install time*: the sum of the time spent on all screens up to, and including, the agreement screen. In the summary condition, this includes the summary, welcome, and agreement screens. For all other conditions, it includes only the latter two screens. For this measure, we also sum the time spent in these screens across *all* installation runs. While this measure is not perfect (since there is variability in how often people ran an installer more than once, and each run adds to this measure), it helps assess the potential impact of summaries by summing time across screens. We define the *total maximum scroll amount* in the same way. Figures 9 and 10 include summaries of these measures.

An ANOVA indicates no significant differences between treatment conditions for total pre-install time, though there is an apparent trend ($p = 0.089$). This trend appears to be due to both the moderate and heavy conditions. This hypothesis is supported by an ANOVA of scrolling behavior across all installation runs (rather than just the first run of each installer), which shows significance ($F_{4,76} = 3.36$, $p = 0.026$). Post-hoc analysis indicates differences between the heavy and summary condition ($p < 0.05$) and the moderate and summary condition ($p < 0.05$).

5.7.3 *Preferences*

Subjects were asked to rate the visual appeal of the software agreements. An ANOVA indicates a significant difference in visual appeal between conditions ($F_{4,76} = 7.61$, $p < 0.001$), with post-hoc analysis revealing significant differences between heavy and control ($p < 0.01$); heavy and summary ($p < 0.01$); moderate and control ($p < 0.05$); and moderate and summary ($p < 0.05$). The minimal condition was also found more appealing than the summary condition ($p < 0.05$) and trended towards being more appealing than the control condition ($p = 0.10$).

5.7.4 *Qualitative Feedback*

In the questionnaire, subjects were asked to provide qualitative feedback regarding the visual appearance of the software agreements and their overall informativeness. A number of comments suggest the techniques worked as intended. For example, a participant in the heavy condition commented:

> *It got me to read them, when I install other programs, I NEVER read them. Big letters, organized points, and cartoons help. I think the organization was the most important.*

A participant in the moderate condition suggests the effectiveness of the documents' pacing:

> *Compared to other agreements on other programs, this one is appealing because of the headings; which are placed similar to a newspaper to get one's attention. On other programs, it is just a bunch of words bunched together, similar to a contract but on the monitor.*

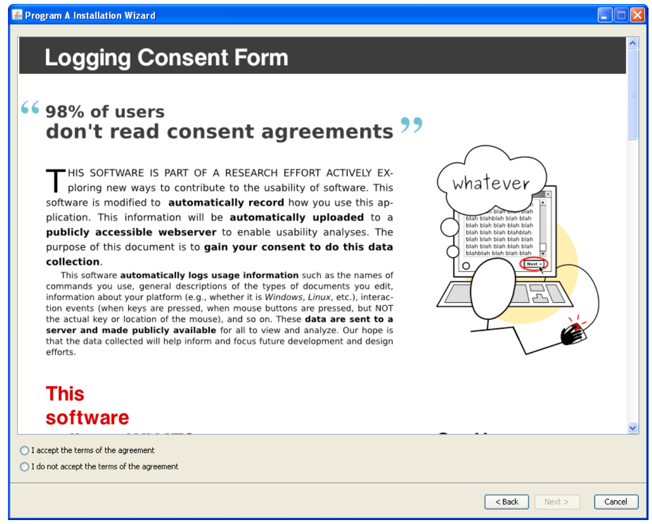


Figure 11. Consent screen showing a textured agreement from the experimental condition in experiment 2. This is the same agreement shown in Figure 1.

Comments from subjects in the minimal condition indicate they noticed the changes in visual appearance. However, they did not comment on the organization of information or the notion of being compelled to read, as in the moderate and heavy conditions.

While the techniques were generally successful, a few comments indicated that not everyone found the redesigns visually appealing. For example, a subject in the heavy condition wrote:

> *I think I would have read it more closely if it had been a little less over-the-top. I did really like that it was different and caught your attention.*

Another subject noted that the heavy style was "somewhat obnoxious in coloration and layout." These comments suggest that while the heavy application of techniques attracted attention for some, it may be too much; the moderate application may strike a better balance.

5.8 Summary

In this experiment, we found increased reading times in the moderate and heavy conditions. The lack of an observed increase in the minimal condition hints that the techniques have an effect beyond the initial impact of the design; however, we wanted to determine the extent of that effect on participants' retention of agreement content. We designed a second experiment to focus on this question.

6. EXPERIMENT 2

6.1 Experimental Design

As in the first study, we conducted a between-subjects deception experiment, but with only two conditions: a textured agreement and a plain-text control condition. Subjects were asked to download, install, and use a single image manipulation application (as opposed to three in the first experiment). The same instrumented installation environment was used as before. However, the distractor task of using the application was not actually performed by participants (though the instructions asked them to use the software after installing it to rate its usability). Instead, participants were interrupted after reaching the point in the software installation process where the software would actually be installed. Instead of installing the software, the participant was stopped and given a content quiz to test how much information they absorbed from the agreement process. This approach minimized the time between exposure to the agreements and taking the quiz.

6.2 Procedure

As before, subjects were given a written scenario and instructions after obtaining verbal consent. The scenario indicated that they had recently received a digital camera, but lacked software to perform basic manipulations of the images. Accordingly, they were told to imagine they had just found the website of an image manipulation application. The instructions asked them to download, install, and evaluate the application, and to decide whether they would continue to use this program on their home computer. Once they had reached a decision, the instructions indicated that they would be given a questionnaire.

After receiving and reading the instructions, subjects had the opportunity to ask questions. They were then seated at a desktop computer with a web browser already opened to the download page of the application, called "Program A". Participants were then able to download and run the installer. After clicking the "Next" button on the software agreement screen in the installer, a full-page screen informed participants that the task portion of the study was complete, and that the researcher will set up the questionnaire (this screen had no visible controls, to prevent participants from accidentally or instinctively skipping past it).

The same VMWare setup and installation apparatus from the first study was also used in this experiment.

6.3 Experimental Conditions

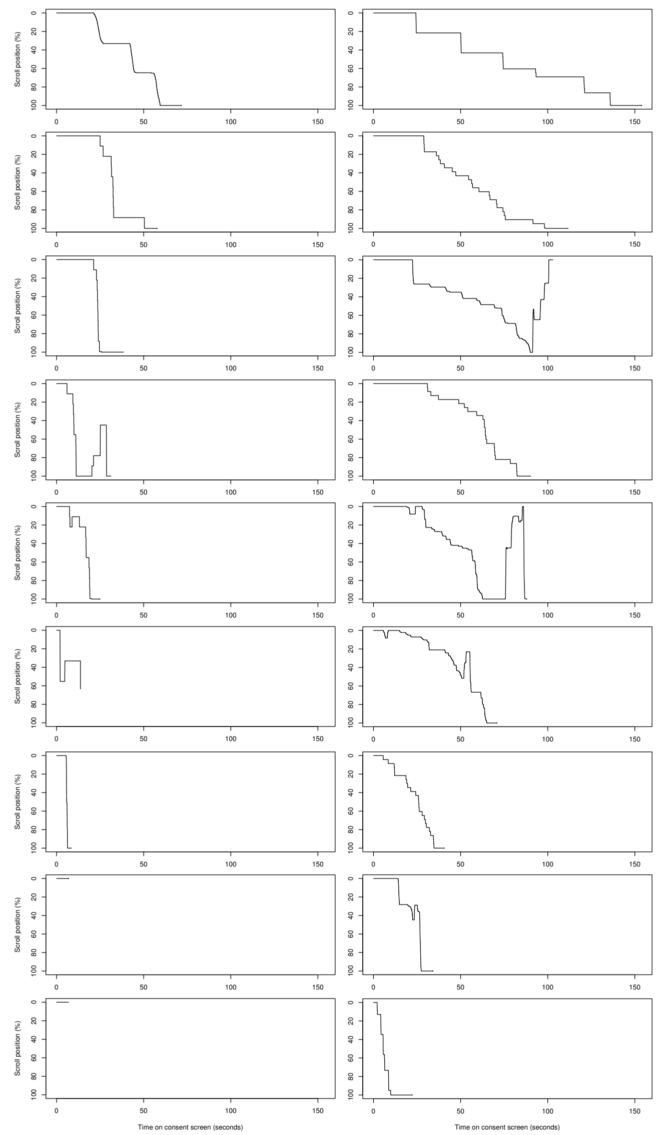
This experiment had two conditions: control and textured. An agreement from experiment 1 was used as the plain-text control.

In the first experiment, the lack of an increase in reading time for the minimal condition emphasized the need for a careful application of our techniques. Using feedback from the first experiment, we further iterated on the design of the textured agreements. We developed a single design that makes more use of vignettes and sensationalism than our moderate condition, but which is more judicious than the heavy condition in their use (see Figure 11). The improved design, applied to the same text as in the control condition, forms the textured condition for this experiment.

6.4 Performance Measures

As before, we measure the amount of time spent on individual installer screens and record the agreement scroll position over time. Since this experiment has only one application, and users are interrupted immediately after completing the agreement screen, we can measure the time spent on the agreement screen without needing to consider multiple runs of the installer.

As part of the computer-based questionnaire, an open-ended content quiz was given to participants in order to gauge how much information they retained from the agreement. Participants were given 8 questions of varying difficulty that required them to recall content from the agreement. Each question began with the phrase,



A. Control Participants **B. Textured Participants**

Figure 12. From experiment 2: scrollbar position as a function of time for the 9 participants with the longest reading time in each condition (sorted by reading time). Time elapsed since the consent screen was shown (x axis) is plotted against the position of the scroll bar (y axis).

"According the Program A's license agreement..." Some example questions include:

- According the Program A's license agreement, what characteristics of your images does Program A record?
- According the Program A's license agreement, if you have questions about the software or the study, who can you contact?

Questions were chosen from content in all areas of the agreement. Potential questions were piloted and refined to ensure that the questions could be answered fully if the entire agreement had been read.

Questions were scored out of 2 points each, making a maximum possible score of 16. A score of 2 was given to complete answers, and a score of 1 given to incomplete answers that nevertheless indicated some awareness of the content in question. For example, to the second question listed above (asking who users can contact for more information) one participant answered:

> *The primary investigator, John Smith, via email at smith@***learning.***.*

This answer was given a score of 2. Another participant answered:

> *There was an email address provided but I don't remember what it was.*

This answer was given a score of 1.

6.5 Participants

28 subjects were recruited in a university setting. Subjects were compensated with a $10 gift certificate for a coffee chain. Subjects were screened to ensure all were native English speakers to minimize potential effects arising due to language ability. 17 females and 11 males participated, aged 19-31 years old (mean=24, SD=3). Subjects' self-reported computer expertise on a five point scale was an average of 3.5 (SD=1) with 5 being "most expert."

6.6 Results of Experiment 2

6.6.1 Timings and Scrolling Behavior

As in experiment 1, subjects were asked to self-report their tendency to read software agreements on a five-point scale. We again found this measure of reading habits to be a contributing factor to the scores, and include it as a factor in our analyses.

Table 1 summarizes the time spent on the agreement screen (in seconds) and the maximum scroll distance into the document (as a percentage of document length). As in experiment 1, the textured condition shows longer reading times than control (mean of 53.6 seconds versus 19.7 seconds), and an ANOVA shows these difference are significant ($F_{1,26} = 7.54$, $p < 0.05$). Maximum scrolling distance into the document was also greater in the textured condition (mean of 71.4% versus 47.4%), and these differences were also significant ($F_{1,26} = 7.16$, $p < 0.05$).

In addition to maximum scrolling time, we examined the scrolling patterns of participants. Figure 12 comprises a series of graphs representing the scrolling behaviour for 18 of the 28 participants. Shown are the 9 participants from each of the 2 conditions who spent the most time reading the agreement within their condition (the remaining 5 participants from each condition had similar scrolling behaviour to the lowest ranked participants shown). Participants are arranged in two columns: control condition on the left and textured on the right. Each column is sorted by reading time. From this diagram, it appears that participants in the textured condition are more inclined to read or skim through the entire agreement at a slower pace, while those in the control condition tend to skip directly to the end of the document (though they were not required by the software to do so). Only one or two participants in the control condition exhibit behaviour comparable to the top 6 in the textured condition.

Table 1. Summary of experiment 2 results across conditions.

Measure	Control	Textured
Time on consent screen (seconds)	mean=19.7, SD=22.2	mean=53.6, SD=48.9
Maximum scroll distance (% of document length)	mean=47.4, SD=50.0	mean=71.4, SD=46.9
Open-ended quiz score (out of 16)	median=0, IQR=4	median=4, IQR=5.75

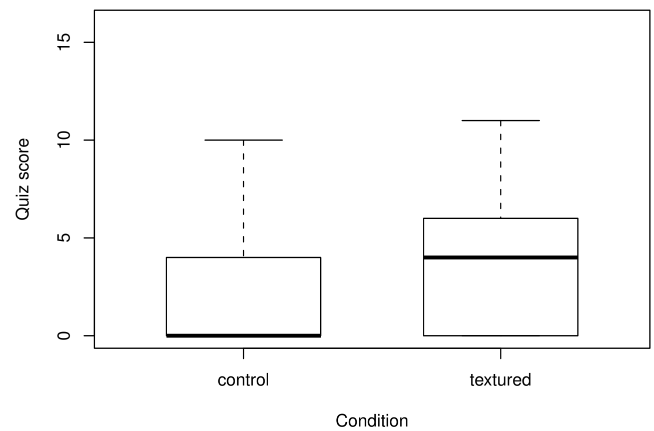


Figure 13. Box plot of quiz scores in experiment 2.

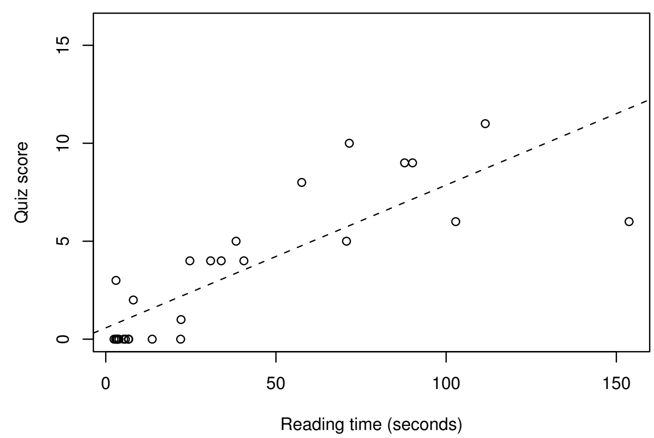


Figure 14. Quiz score versus reading time in experiment 2 with linear regression (r = 0.83).

6.6.2 Content Quiz

Participant scores on the content quiz are also summarized in Table 1. As can be seen in Figure 13, both distributions are heavily skewed; as a result, we believe that median, not mean, is a more appropriate measure of the location of these distributions. The median score in the textured condition was 4 (IQR=5.74), compared to 0 (IQR=4) in the control condition. An ANOVA on ranks suggests this difference is significant ($F_{1,26} = 4.18$, $p = 0.052$). Quiz scores were also found to be highly correlated with the amount of time participants spent reading the agreement ($r = 0.83$, $p < 0.0001$)—see Figure 14.

6.6.3 Qualitative Feedback

As in experiment 1, qualitative feedback was collected in a questionnaire and a post-task interview. In particular, participants were asked why they read less, more, or the same amount of content in comparison to their normal behaviour when installing software. Participants in the control condition all responded that they read the same amount as usual, and often made comments such as, “All of the EULAs are basically the same,” or, “The same amount was read, since they generally all say the same things.”

By contrast, a number of participants in the textured condition stated they read more content than usual, and cited elements of the agreements that pulled them in; for example:

> *I read more due to the diagram stating that 98% of the users don't read the License Agreement.*

Both in written responses and during interviews, participants in the textured condition noted being pulled in by prominent elements at the top of the document—such as the pull-quote mentioned in the above quotation (Figure 5), or a vignette serving a similar purpose (Figure 6)—and then were compelled to continue reading further into the document. These findings provide evidence that such attention-grabbing embellishments can pull participants into the content of the rest of the document.

7. DISCUSSION

The results of both experiments support the notion that textured agreements compel people to engage with the agreements more than with plain-text agreements. In experiment 1, textured agreements increased the time spent on agreement screens from an average of 7 seconds in the control condition to an average of 36-40 seconds in the moderately and heavily designed treatments. Notably, these effects were not observed in the minimal condition, suggesting this increase in time cannot be attributed to novelty alone.

Experiment 2 saw a similar increase in reading time between conditions of 34 seconds (from 20 to 54 seconds). Longer reading times were also found to correlate strongly with higher quiz scores. This, combined with the higher scores in the textured condition, suggests that textured agreements are successfully capturing and retaining attention long enough for users to absorb more information from the agreements. That users are engaging more with the textured agreements is also evident in the change in reading behaviour seen in Figure 12: more subjects in the textured condition appear to be spending time looking through the *entire* document than in the control condition.

The success of these techniques in increasing reader engagement is promising for two reasons. First, the techniques employed represent only a subset of the wide repertoire of visual design techniques available; there is significant room for further improvement. Second, the technique achieved its effectiveness by operating on the *primary object of interest*, namely, the software agreement itself. As we discuss next, there are good reasons to consider improving this document rather than introducing auxiliary documents.

7.1 Assessing the Summary Condition

In experiment 1, the summary condition was not shown to significantly affect the reading time of the actual software agreements. However, this finding does not indicate that summaries are ineffective. A comparison of time spent on the summary screen to time spent on the welcome screen of the other conditions indicates that users *do* read the contents of the summary screen, spending 18 seconds longer on this screen compared to the welcome screen of the other conditions ($p < 0.0001$) (see Figure 9). However, our data indicate that there are side effects associated with reading the summary. In particular, the data argue that summaries reduce the likelihood that people spend time reading the *actual* software agreement. This echoes findings from a previous study by McDonald *et al.* [19] that examined participants' accuracy in a multiple-choice quiz on the content of privacy policies with different presentations. That study found that participants were significantly less accurate for summary agreements than plain text agreements when the answer to a question was not present at the summary layer. As hinted at in our study, participants were less likely to read into the full text to find the answer. Thus, while summaries have the potential to effectively communicate a condensed version of the agreement, they do so at the cost of reading the full agreement.

7.2 Human Ethics Perspective

Having observed the positive effects of the textured agreements, we met with three members of our internal review board who regularly review human ethics applications. We presented the textured agreements to gain their perspectives and understand potential issues in using them in practice.

The reaction to the agreements was extremely positive. Compelling study participants to read consent agreements is a problem they struggle with in study designs, so they welcomed the visual redesigns. However, they did have some suggestions for improving the designs and for future research. In particular, they observed that the heavily textured agreements could be problematic for seniors. This population might find the dense clustering of information distracting or difficult to comprehend. This point raises an important issue for future work: examining potential age differences related to the particular designs.

8. CONCLUSION AND FUTURE WORK

This paper has introduced textured agreements, visually redesigned software agreements that draw upon strategies used in other visual media to gain users' attention, retain that attention, and highlight information of personal relevance. Our studies suggest that these techniques show promise in improving the software agreement process. Our results also suggest caution in using techniques that partition the agreement process into multiple phases, as is done with summaries of agreements. While summaries are effective at conveying a synopsis of the agreement, they can lead users to ignore the full agreement.

While the results of the minimal condition in the first experiment show that more than novelty is required to engage readers, we would like to perform longer-term evaluations of textured agreements to determine their robustness with respect to desensitization and habituation. Additionally, we wish to study different demographic groups, particularly seniors, to understand how these techniques affect readability and reading behaviors. Finally, we would like to apply these techniques to a number of real-world agreements to better understand their reproducibility and adaptability. To that end, we have developed templates to aid in others' adoption of these techniques, available from our website[1].

1 http://hci.uwaterloo.ca/research/textured.

9. REFERENCES

[1] Abrams, M., Eisenhauer, M. and Sotto, L. (2004). *Response to the FTC request for public comments in the Advance Notice of Proposed Rulemaking on Alternative Forms of Privacy Notices under the Gramm-Leach-Bliley Act*. Center for Information Policy Leadership.

[2] Amer, T. S. and Maris, J. (2004). *Signal Words and Signal Icons in Information Technology Exception and Error Messages*. Northern Arizona University.

[3] (2008). *Aviary Terms of Use*. Aviary. http://aviary.com/terms. Last checked April 5, 2009.

[4] Brustoloni, J. C. and Villamarin-Salomon, R. (2007). Improving security decisions with polymorphic and audited dialogs. *SOUPS '07*. 76–85.

[5] Campbell, F. A., Goldman, B. D., Boccia, M. L. and Skinner, M. (2004). The effect of format modifications and reading comprehension on recall of informed consent information by low-income parents: a comparison of print, video, and computer-based presentations. *Patient Education and Counseling*. 53: 205–216.

[6] Egelman, S., Tsai, J., Cranor, L. F. and Acquisti, A. (2009). Timing Is Everything? The Effects of Timing and Placement of Online Privacy Indicators. *CHI '09*.

[7] Frascara, J. (2004). *Communication design: principles, methods, and practice*. Allworth Communications, Inc.

[8] Friedman, B., Lin, P. and Miller, J. K. (2005). Informed consent by design. *Security and usability: designing secure systems that people can use*. Ed. Cranor, L. F. and Garfinkel, S. O'Reilly Media, Inc.

[9] Friedman, B., Smith, I., Kahn, P., Consolvo, S. and Selawski, J. (2006). Development of a Privacy Addendum for Open Source Licenses: Value Sensitive Design in Industry. *Lecture notes in computer science*. 4206: 194.

[10] (2007). *GNU General Public License, version 3*. Free Software Foundation. http://www.gnu.org/copyleft/gpl.html. Last checked May 13, 2010.

[11] Good, N., Dhamija, R., Grossklags, J., Thaw, D., Aronowitz, S., Mulligan, D. and Konstan, J. (2005). Stopping spyware at the gate: a user study of privacy, notice and spyware. *SOUPS '05*. 43–52.

[12] Good, N., Grossklags, J., Mulligan, D. K. and Konstan, J. A. (2007). Noticing notice: a large-scale experiment on the timing of software license agreements. *CHI '07*. 607–616.

[13] Grossklags, J. and Good, N. (2008). Empirical Studies on Software Notices to Inform Policy Makers and Usability Designers. *Financial Cryptography and Data Security*. 341–355.

[14] Hellier, E., Wright, D. B., Edworthy, J. and Newstead, S. (2000). On the stability of the arousal strength of warning signal words. *Applied Cognitive Psychology*. 14: 577–592.

[15] (2003). *HIPAA Highlights Privacy Notice*. Center for Information Policy Leadership, Hunton and Williams. http://www.hunton.com/news/news.aspx?nws_pg=7&gen_H4ID=10102. Last checked Sep 19, 2008.

[16] Jensen, C. and Potts, C. (2004). Privacy policies as decision-making tools: an evaluation of online privacy notices. *CHI '04*. 471–478.

[17] Kelley, P. G., Bresee, J., Cranor, L. F. and Reeder, R. W. (2009). A "nutrition label" for privacy. *SOUPS '09*. 1–12.

[18] Kress, G. R. and Van Leeuwen, T. (2006). *Reading images: the grammar of visual design*. Routledge.

[19] McDonald, A., Reeder, R., Kelley, P. and Cranor, L. (2009). A Comparative Study of Online Privacy Policies and Formats. *PETS '09: Proceedings of the 9th International Symposium on Privacy Enhancing Technologies*. 37–55.

[20] Patrick, A. S. and Kenny, S. (2003). From privacy legislation to interface design: Implementing information privacy in human-computer interactions. *Lecture Notes in Computer Science*. 2760: 107–124.

[21] (2007). *Ten steps to develop a multilayered privacy notice*. Center for Information Policy Leadership, Hunton and Williams. http://www.hunton.com/files/tbl_s47Details%5CFileUpload265%5C1405%5CTen_Steps_whitepaper.pdf. Last checked May 13, 2010.

[22] Terry, M., Kay, M., Vugt, B. van, Slack, B. and Park, T. (2008). Ingimp: introducing instrumentation to an end-user open source application. *CHI '08*. 607–616.

[23] Tufte, E. R. (1995). *Envisioning Information*. Graphics Press.

[24] White, J. V. (1982). *Editing by design*. Bowker New York.

[25] Wogalter, M. S. and Laughery, K. R. (1996). WARNING! Sign and Label Effectiveness. *Current Directions in Psychological Science*. 5: 33–37.

[26] Wright, P. (1977). Presenting technical information: a survey of research findings. *Instructional Science*. 6: 93–134.