Communicating Software Agreement Content Using Narrative Pictograms

Matthew Kay
HCI Lab
University of Waterloo
200 University Avenue West
Waterloo, Ontario, CA, N2L 3G1
mjskay@cs.uwaterloo.ca

Michael Terry
HCI Lab
University of Waterloo
200 University Avenue West
Waterloo, Ontario, CA, N2L 3G1
mterry@cs.uwaterloo.ca

Abstract
We present *narrative pictograms*, illustrative diagrams designed to convey the abstract concepts of software agreements. Narrative pictograms arose out of a need to create software agreements that are comprehensible without written language. We first present example diagrams designed to describe the data collection policies of research software, and the composition rules used to create them. We then present our design process and lessons learned during design. Finally, we present results from an evaluation based on the ISO 9186-1 test for graphical symbols.

Author Keywords
Informed consent, pictograms, wordless diagrams, open source

CHI 2010, April 10–15, 2010, Atlanta, Georgia, USA.
ACM 978-1-60558-930-5/10/04.

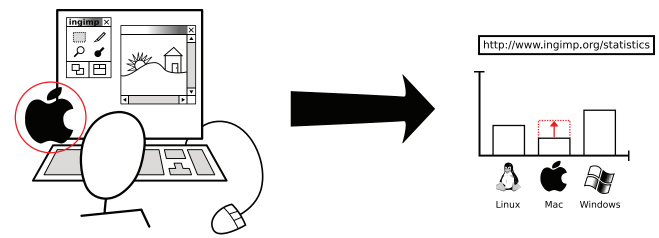


Figure 1. A narrative pictogram. This diagram illustrates the fact that the software will collect data about which operating system the user uses.

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

ACM General Terms
Design, Experimentation, Human Factors.

Introduction
End users must often agree to lengthy terms of use prior to installing software or using software services. These terms of use may take the form of End User License Agreements (EULAs), privacy policies, or, in the case of computer-based human subjects research, consent agreements. While these various agreements each serve slightly different purposes, we collectively refer to them as *software agreements*.

Currently, software agreements are communicated primarily via text. These documents typically take significant time to read (often 20 minutes or more) and usually require a university or postgraduate reading level [6]. As a result, these agreements are rarely read [2]. A number of methods have been developed to increase the chance that individuals actually take the time to read information presented, or to improve comprehension of that content. For example, Good *et al.* [2] demonstrated that summarizations of EULAs can increase the chance that some terms of use are read and acted upon; similarly, Kelley *et al.* [7] took inspiration from nutrition labels to design improved summaries of website privacy policies.

Existing techniques require the ability to read and comprehend the language in which the agreement is written. However, not all software is localized for every language, meaning that users may encounter software agreements in languages other than their native tongue. This presents a significant barrier to gaining truly informed consent, which is especially problematic for researchers conducting studies over the Internet. As a result, both users and software producers would benefit from techniques that improve the ability to effectively communicate agreement terms to a linguistically diverse audience.

The success of wordless communication in other contexts suggests it may be a promising approach to reducing localization requirements. Pictograms (compact graphical symbols representing a single concept or object) have been highly successful in informational signs and road signs, such as Otl Aicher's 1972 Munich Olympics signage or the symbols commissioned by the US Department of Transportation to communicate transportation services [10]. More elaborate wordless diagrams have been employed successfully in instructional manuals, such as the "welcome mats" created by Patrick Hofmann for HP [3,4]. These welcome mats improved usability while saving HP hundreds of thousands of dollars in printing costs [4]. In the HCI community, recent work by Medhi *et al.* has shown that text-free user interfaces can communicate effectively to illiterate populations [9]. Similarly, McGrenere *et al.* have explored the use of interfaces relying more heavily on visual and auditory cues to assist those with aphasia [8].

Building on these successes, we introduce *narrative pictograms* (Figure 1), a pictorial technique specifically designed to supplement (not replace) text-based software agreements in order to better communicate agreement terms to a diverse population of users. In the rest of this paper, we outline this technique as applied to illustrating the data collection policies of a publicly distributed research application. We then describe results from formative and summative evaluations of the diagrams, and conclude with directions for future work.

Narrative Pictograms of Data Collection

In this section, we first describe the specific communication goals that motivated the creation of narrative pictograms. We then present the full technique and its corresponding composition rules, with illustrative examples.

Goals of the Illustrations

This work began with the need to create a consent agreement to accompany a research application, *ingimp* [11], which collects software usage data (e.g., the commands people use). Our specific goals were to:

- Augment, not replace, the existing text-based consent agreement.

What is ingimp?

ingimp [11] is a publicly available version of an open source bitmap graphics editor (the GIMP [1]) that we have modified to collect software usage data. Since its release, we have amassed a user base comprising nearly a dozen different locales.

ingimp includes a text-based consent agreement that outlines the data it collects and the ways a participant could be placed at risk by using the software. However, this consent agreement is written in English, and we have limited resources for localization; consequently, we face a real challenge in gaining (truly) informed consent from users.

See *http://www.ingimp.org/*

- Increase the chance that the user understands the purpose and functionality of the software, specifically, the types of data collected and not collected.

- Reduce the need to localize the diagrams themselves: the diagrams should contain minimal text and, to the extent possible, one should be able to learn how to read the diagrams without the need of auxiliary aids.

To achieve these goals, we engaged in an iterative design process, discussed later. Here we describe the resultant diagrams and the composition rules used to produce them.

Narrative Pictograms of Data Collection

The narrative pictograms we developed are segmented into four collections of diagrams, each serving a different purpose in describing the data collection policies of the instrumented software. Users see these in the order outlined below to create a narrative that teaches them how to read the diagrams; this narrative gradually instructs users on what data are, and are not, collected. The four sets of diagrams are:

1. *The Functional Overview*. An illustration that provides a functional overview of how the software works. This overview establishes a context for the narrative.

2. *Environmental Data Collection*. These illustrations show what data are logged without explicit user action (e.g., the user's operating system, their locale, etc).

3. *Interaction Data Collection*. These diagrams show what data are logged as a result of explicit user action (e.g., interacting with windows).

4. *Privacy Sensitive Data Collection*. These illustrations depict data collected that may be considered private or sensitive in nature.

With the exception of the functional overview, each set of diagrams is comprised of multiple individual diagrams, each of which corresponds to a single concept or scenario of use. In contrast, the functional overview is a single, larger illustration that conveys an establishing narrative. We describe each set of diagrams in detail below.

Functional Overview

The first illustration displayed in our narrative pictograms is the functional overview (Figure 2). The functional overview depicts a basic interaction sequence from the time the application is started to when it is closed. The basic use of the application is complemented with graphs suggesting data collection: as the user in the illustration interacts with the mouse or inputs a filename, bar graphs corresponding to these inputs are incremented proportionally. When the application is closed, the user's personal bar graphs are shown being transmitted to the research website.

The functional overview teaches the user how to read the diagrams by presenting a concrete, simplified series of familiar actions. This sequence establishes a context in which more novel elements can be introduced. The conventions and visual elements established in this overview are repeatedly used in subsequent diagrams and, when necessary, purposefully violated in order to draw attention to important information. This basic strategy permeates the diagrams' design: a "familiar" scenario is presented to the viewer, with a select set of novel elements introduced into that scenario. Over time, repeated use of novel elements turns them into known, familiar

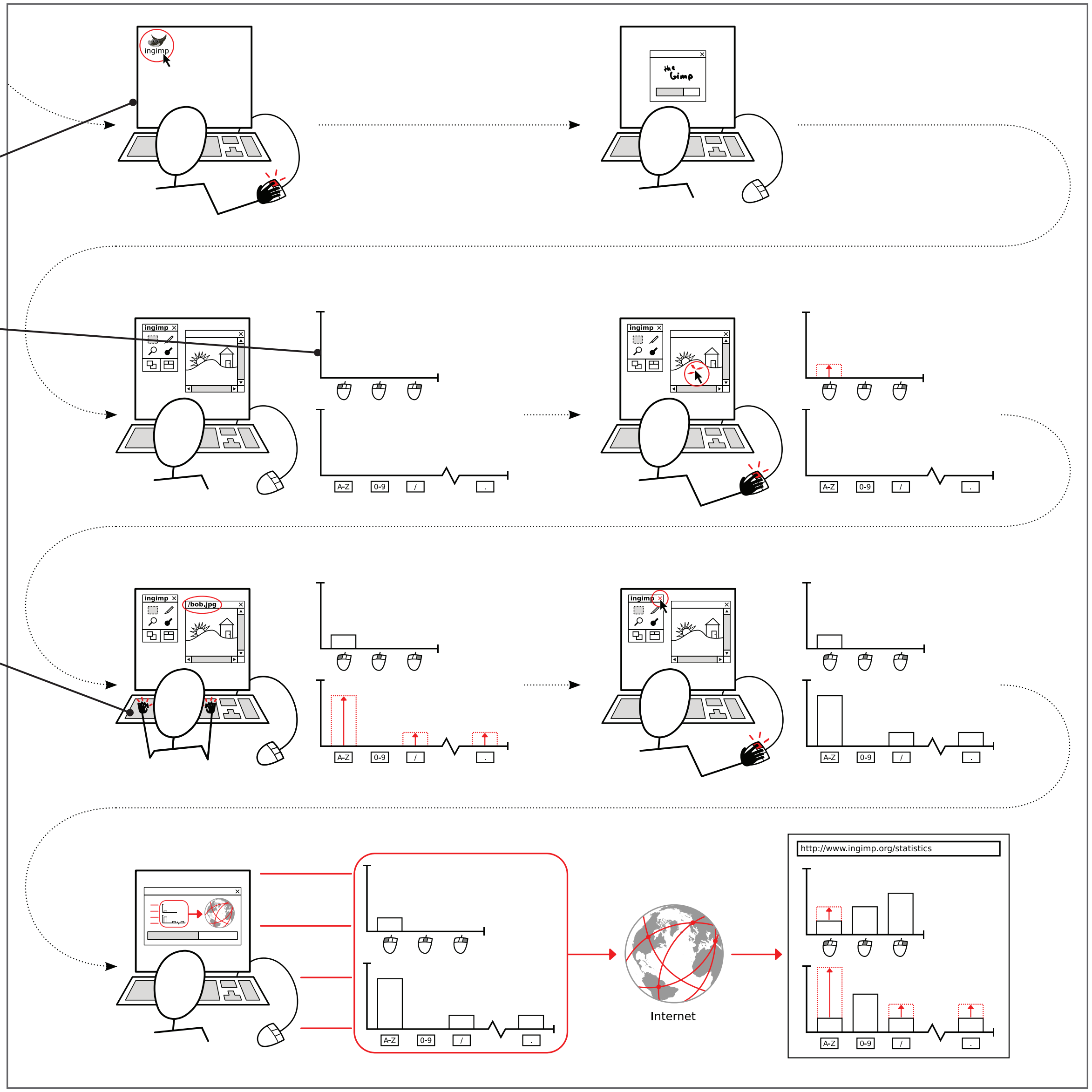
Figure 2. The functional overview provided to describe the fact that the ingimp software collects usage data that is sent to a website.

We begin by establishing a concrete context: the launch and use of an application.

We then introduce an easily-identified anomoly: graphs (initially blank). The purpose of these graphs may not be immediately obvious to many readers; this is expected.

Next, we clarify the meaning of the graphs through two concrete examples: a mouse click and (in this example) typing. We highlight each action and its consequences (the data collected) to help the reader understand what the graph represents. Pictures of the graph's source data are used as labels to improve the correspondence between the character's actions and the graphs.

Finally, we depict the transmission of the collected data—the graphs—to our servers, via the Internet.



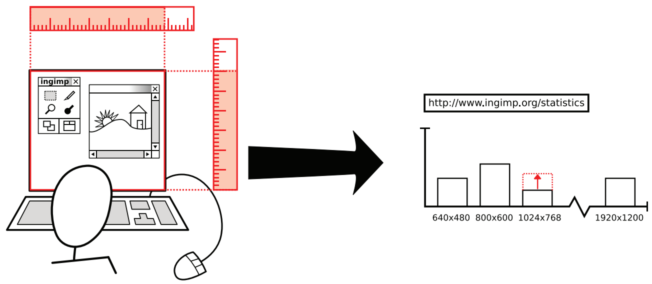


Figure 3. A diagram in the "environmental data collection" set, depicting the capture of screen resolution. Graph labels with common screen resolutions reinforce that monitor resolutions are recorded.

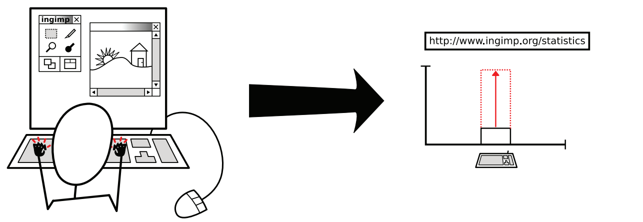


Figure 4. A diagram in the "interaction data collection" set, depicting the fact that keyboard activity is recorded. By using a single bar in the graph, most viewers correctly assume that individual keystrokes are not recorded.

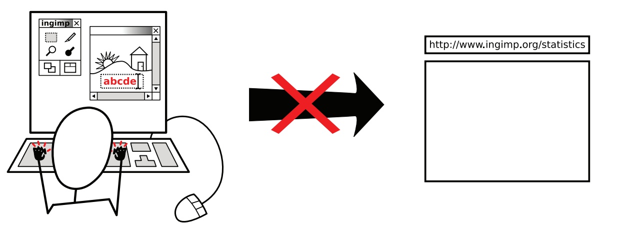


Figure 5. This diagram immediately follows that in Figure 4 to emphasize that individual keystrokes are not collected. This is done by breaking previously established conventions for data collection: the arrow and graph.

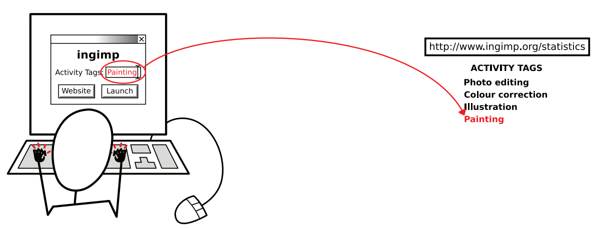


Figure 6. This diagram also breaks conventions established in previous diagrams (the large arrow) to highlight data that are recorded without any summarization or anonymization: in this case, activity tags.

elements, allowing further augmentation of scenes with other, novel elements. In this way, the viewer is incrementally introduced to increasingly sophisticated concepts.

The next three sets of diagrams build on the conventions established in this overview to represent what data are, and are not, collected. They accomplish this goal by repeating the same motif developed in the overview: a user shown at the computer with important actions and concepts highlighted in red, and a summarization of the data collected shown in a data collection graph.

Environmental Data Collection

The second set of narrative pictograms depicts "environmental data" collected, or data that describe a user's task environment. For example, the diagrams illustrate that the user's operating system and the size of their monitor are logged (Figures 1 and 3).

In contrast to the functional overview, which utilized a narrative composed of a string of related scenes, this series of illustrations reduces the depiction of logging to single, independent scenes. As mentioned, the character is shown in front of the computer with the data being collected highlighted in red. An arrow placed between the user and the data collection graph suggests the collection and transmission of the data to a remote website (Figures 1 and 3). In this series, the data collected are not data that result from specific user action (such as the user's time zone). Accordingly, the user is shown sitting in front of the computer with no hands visible.

Like the graphs shown in the functional overview, the data collection graphs have very literal graph labels. For example, icons for popular operating systems are used

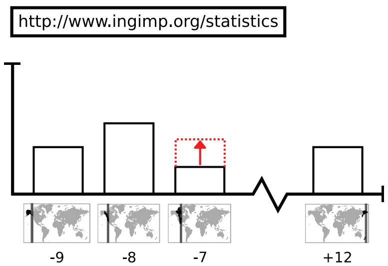


Figure 7. Cropped view of a diagram showing the collection of time zone data. As time zones allow a relatively familiar, concrete depiction—shaded world maps—this diagram is placed early in the Environmental Data Collection set.

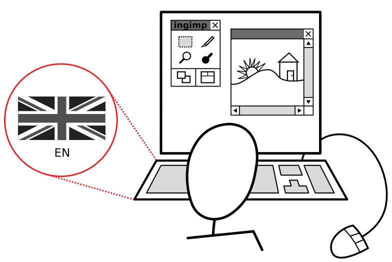


Figure 8. Cropped view of a diagram showing the collection of locale information. This more abstract concept is introduced later in the Environmental Data Collection set.

to indicate that the type of operating system is logged (Figure 1). Similarly, common screen resolutions (e.g., 1024x768) are used as labels to indicate the capture of screen resolution (Figure 3).

The order of illustrations within this series follows our strategy of first showing objects and concepts that are most likely to be familiar and known to users. Thus, the first scene depicts the logging of the user's operating system (using familiar icons), followed by time zone data using a shaded world map (Figure 7). The collection of locale information, a slightly more abstract concept, is shown after these more concrete concepts, at a point when the user is more likely to understand the purpose of the individual diagrams. We represent locale data using countries' flags associated with the keyboard (Figure 8).

Interaction Data Collection

The next series of diagrams, interaction data collection, depicts scenes in which an explicit user action results in some data being collected. For example, the system records that the mouse or keyboard was used, so use of the mouse and keyboard is shown, alongside appropriate data collection graphs. As in the previous set, we start with concepts more likely to be familiar: mouse clicks, keyboard use, and tool use (Figure 9).

For this particular application, the actual mouse location and specific keystrokes are not logged (just the fact that the mouse or keyboard was used). To make these points clear, we alter a number of elements in the previously established visual motif. As an example, the keyboard data collection diagram (Figure 4) shows the user typing on the keyboard, but its data collection graph displays only a single bar labeled with an icon of an entire keyboard. This depiction suggests that all of the keyboard activity is summarized in one dimension only, since individual bars for each key are not shown. Our formative study revealed that the absence of information within a diagram suggests the absence of that data being collected by the application, making this a useful strategy to help convey this concept.

The ability to infer the absence of data collection is useful, but we also wanted to make it explicit that certain types of data are not collected. To make this fact clear, the diagram in Figure 5 is shown immediately after the diagram in Figure 4 to clarify that the actual, typed text is not recorded. For this class of diagram, we modify the typical data collection convention by placing a red "X" through the data transmission arrow and showing a blank website. Having already shown several examples of data being collected and represented on the website, these two modifications to an established convention help communicate that data are *not* transmitted.

Privacy Sensitive Data Collection

The previous two diagram sets employ the convention of showing data being collected and aggregated with other data, suggesting a level of anonymity to the data collection process. However, some data collected by this software is recorded without any summarization or anonymization applied. For example, our software allows users to describe their planned use of the software by entering keywords in a special box shown at start-up. These keywords—referred to as *activity tags*—are logged without modification, and are publicly accessible (as is all other data collected by this software).

To highlight the sensitive nature of this type of data collection process, we break a previously established

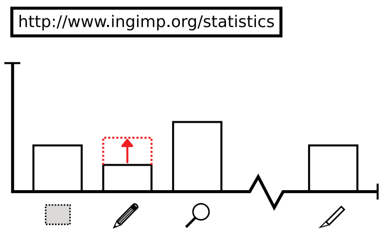


Figure 9. Cropped view of a diagram showing the collection of tool use. This diagram makes use of concrete labels drawn from ingimp: tool icons.

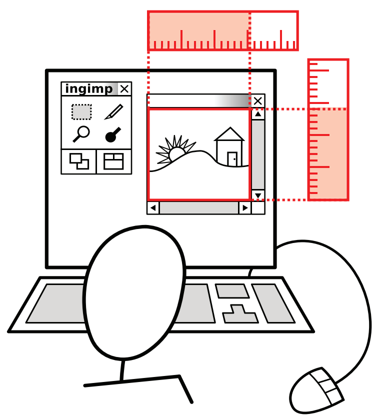


Figure 10. Cropped view of a diagram showing the collection of image size. This diagram reuses the visual convention for "measurement" previously established in the diagram depicting the collection of screen resolution (Figure 3).

convention—the large arrow indicating data collection—and instead show data being taken directly from the user's screen and placed into the public website's data set (Figure 6). This direct correspondence between information on the character's screen and the public website underscores the associated privacy risks.

Having described the four sets of narrative pictograms developed for our specific application, we now describe the composition rules employed to create these diagrams.

Composition Rules

The composition rules are as follows:

- Employ literal contexts and simplified representations of objects, rather than abstract symbols
- Use repetition to establish patterns and conventions
- Break established patterns and conventions to draw attention to details
- Avoid conventions of manuals to avoid creating the impression that the diagrams are instructional
- Avoid reliance on domain knowledge, or provide additional context for complex domain-specific concepts

These composition rules, in and of themselves, are not necessarily novel: graphic designers and technical illustrators regularly apply them in various contexts. However, it is the careful selection and application of these techniques to this particular problem domain that results in a novel contribution. We expand on each rule in turn.

EMPLOY LITERAL CONTEXTS

Narrative pictograms are presented to users in the highly unique context of a software agreement process (e.g., during software installation). They are also intended to function across cultures, without the need for localized text to assist in comprehension. These factors require care in the design process so that the intent of these unfamiliar and unexpected illustrations is not misinterpreted. Accordingly, the diagrams employ very literal narratives, with familiar objects and actions, to establish a grounding context. The narratives also help establish the fact that the diagrams are informative and not tutorials or advertisements. With this context established, more abstract concepts can be introduced to communicate more sophisticated material.

USE REPETITION TO ESTABLISH PATTERNS AND CONVENTIONS

Hofmann, in writing about methods to create cross-cultural informational illustrations [4], suggests starting instructions with a base illustration that is gradually changed. The repetitive use of the same base illustration, with only minor modifications, enables the reader to more easily establish what is changing. This same convention is used within narrative pictograms.

Repetition in our illustrations works in several complementary ways. First, by repeating elements without change, we indicate that the repeated element is not very significant; simultaneously, elements that *do* change are more easily identified and granted significance. Second, repetition helps reinforce the concepts being conveyed. For example, the convention of data being transmitted from the person's machine to the public website is used repeatedly. If a user does not initially understand this convention, as they move through the diagrams they will be repeatedly exposed to it, giving

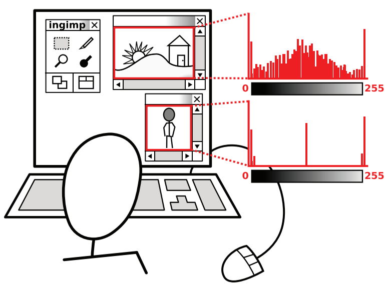


Figure 11. Cropped view of a diagram showing the collection of image histogram data. Many participants had difficulty with this illustration due to a lack of domain knowledge.

them a number of opportunities to interpret it in different contexts. Once they finally understand the convention, they can easily recognize that all of the diagrams are attempting to convey the logging and transmission of data. In our formative evaluation, subjects often exhibited this behaviour, and were able to go back and understand previous diagrams which were initially unclear to them.

Breaking Established Patterns and Conventions to Draw Attention to Details

Once a pattern has been established, it can be broken or modified to draw attention to particular details, or to create new conventions. For example, the diagrams break conventions to highlight notable exceptions of what data are, and are not, collected. As previously discussed, the convention of an arrow used to indicate data collection is broken by some diagrams to indicate data not collected (Figure 5) and data collected without summarization (Figure 6). Both of these instances establish new conventions that are used in subsequent diagrams.

Avoid Conventions of Manuals

Our testing of the diagrams revealed that the use of many conventions from instructional illustrations (e.g., assembly instructions) would immediately lead readers to consider the illustrations to be a user manual.

As an example, our earliest illustrations used numbers in a variety of contexts to connote sequences. We originally numbered the sequences of the functional overview to emphasize the order in which they should be read (since reading direction is not universal). However, this numbering suggested that the diagrams were trying to instruct the viewer on how to accomplish some task. Consequently, we adopted other strategies to suggest sequences (e.g., the use of arrows in Figure 2).

When originally designing illustrations to describe what data are not collected, we placed a prohibition sign (a circle with a line through it) over an illustration showing data collection. Like the numbering, subjects felt they were being instructed, but in this case, instructed *not* to do something. To avoid this connotation, we instead place a red "X" over the arrow to indicate that transmission does not occur (Figure 5). By using an "X" instead of the prohibition sign, and by placing it over the arrow rather than the activity, we avoid the suggestion that we are prohibiting certain user actions.

Avoid Reliance on Domain Knowledge

Our summative evaluation identified an additional issue with a subset of the diagrams: overreliance on domain knowledge. Consider Figure 11, which is intended to depict the collection of image histogram data. These histograms are commonly employed by professional users of photo manipulation applications. However, the histograms may not be understood by novices to the domain. In our summative evaluation, some participants were provided with text from the ingimp agreement, which describes that image histograms are generated from frequency counts of pixel values. These participants were able to acquire the necessary knowledge to interpret the diagram. In contrast, participants without that text expressed confusion about the meaning of the histograms. This finding suggests that more complex, domain-specific concepts require additional written or pictorial context to be understood by relative newcomers to the domain.

Evaluation

The composition rules described above are a result of both formative and summative evaluations. We briefly describe each evaluation.

What is ISO 9186-1?

ISO 9186-1 is a test method for graphical symbols [5]. It prescribes two different types of tests when evaluating graphical symbols: a comprehension test and a judgment test.

In the comprehension test, the subject is presented with a context in which they might see the sign or symbol. They are then shown a symbol and asked to interpret its meaning in that context. This test is meant to measure the likelihood that a symbol will be correctly interpreted by the members of a population.

In the judgment test, the subject is informed what the intent of the symbol is. They are then asked to estimate how many people are likely to understand it in the given target population. This test is meant to assist in selecting the most effective designs among a set of design alternatives.

To avoid possible confounds caused by second-language participants being unable to adequately express their interpretation of a diagram, ISO 9186-1 requires the tests be administered in each participant's native language.

Design Process and Formative Evaluation

We designed the diagrams through an iterative process that incorporated the ISO 9186-1 test method for graphical symbols [5], a thinkaloud protocol, and Wizard-of-Oz prototyping. More specifically, we presented users with paper-based prototypes and indicated that the illustrations would be seen when starting our software for the first time. We then asked subjects to interpret the illustrations for us, thinking aloud. We took notes on their comments and behaviour as they did so. We did not provide any assistance, instructing them to say "I don't know" if they could not understand an illustration. When significant communication failures occurred with the illustrations, we sketched new, alternative illustrations and asked the participant to interpret the new designs.

After each subject was done interpreting the illustrations, we described the illustrations' meanings and asked the subject to estimate the percentage of users who would understand them. This step allowed us to gauge the effectiveness of the designs, and guided the selection and refinement of the designs for the next participant.

We recruited 14 subjects, 7 native English speakers and 7 non-native English speakers; 4 were female and 10 were male. All subjects were university students.

For the first 4 subjects, our initial designs failed to effectively convey the intended concepts of the software agreements. These designs underwent significant modifications in the presence of the subjects and in between trials. After these first four subjects, we converged on a set of designs and a uniform set of composition rules (the first four rules outlined earlier). These designs were iteratively refined with the last 10 subjects of this study. These last 10 subjects were each able to correctly interpret all but 1 or 2 of the resulting 20 diagrams, with no single diagram consistently misinterpreted.

Summative Evaluation

To understand the effectiveness of the narrative pictograms produced, we designed a between-subjects experiment based on the aforementioned ISO 9186-1 test method [5], slightly altered for our needs. In particular, we added a "diagrams with text" condition (in addition to the sole "diagrams only" of ISO 9186-1), which places the illustrations inline with a full text agreement adjacent to the text corresponding to each diagram's content. By comparing the comprehensibility of the diagrams without text to a "best case" scenario (when explanatory text is present), we hoped to identify diagrams in further need of refinement.

20 subjects (15 females, 5 males) were recruited in a university setting and compensated with a $10 coffee shop gift certificate. All were native English speakers.

The mean number of diagrams correctly interpreted per participant were 10.0 (SD=5.77) and 13.6 (SD=5.97) for the "diagrams only" and "diagrams with text" conditions, respectively. Median scores were 11.5 (min=0, max=16) and 16 (min=0, max=19). The distributions did not differ significantly, but there was a trend towards difference (Wilcoxon rank sum $W = 27$, $p < 0.1$ two-tailed).

As expected, participants in the "diagrams only" condition did not perform as well as those provided with supplementary text. However, this difference in performance is not large, and can be attributed to a few problematic diagrams, as can be seen in Figure 8, which compares the number of correct responses for each diagram between the two conditions. What is clear from this figure

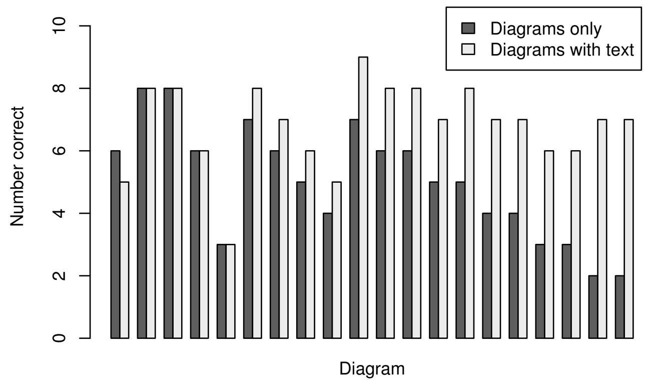


Figure 12. Number of subjects who correctly interpreted each diagram (with 10 the highest possible), ordered by the difference between conditions. Diagrams that significantly under-perform compared to when they are supplemented with written text are further to the right, and may need refinement.

is that much of the difference in performance scores is due to a few problematic diagrams located on the right side of the graph. Written responses and post-task interviews revealed that some of these diagrams relied too heavily on domain knowledge (e.g., Figure 11), leading us to formulate the fifth composition rule discussed above (avoid reliance on domain knowledge).

Limitations and Future Work

Our evaluation provides evidence that the abstract concepts of data collection can be effectively conveyed using wordless diagrams. However, more work is required to further elucidate why certain diagrams under-perform without accompanying text. In addition, research is needed to understand how well our composition rules would generalize to other types of software agreements, such as EULAs.

This paper has presented narrative pictograms, an approach for depicting software agreements using illustrations only. This work is one of the first attempts to comprehensively investigate the problem of localizing consent agreements, a significant problem for commercial software developers, open source software developers, and those conducting research on the Internet. The lessons learned and the composition rules derived from this work provide a foundation for future exploration of this problem space.

References

[1] GIMP. http://www.gimp.org/.

[2] 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. In *CHI '07*. 607–616.

[3] Hofmann, P. (1998). Away with words! How to create wordless diagrams. In *IPCC '98*. 437–438.

[4] Hofmann, P. (2007). Localising and Internationalising Graphics and Visual Information Commentary. In *IEEE Transactions on Professional Communication*.

[5] ISO 9186-1. (2007). *Graphical symbols—Test methods—Part 1: Methods for testing comprehensibility*.

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

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

[8] McGrenere, J., Davies, R., Findlater, L., Graf, P., Klawe, M., Moffatt, K., Purves, B. and Yang, S. (2003). Insights from the aphasia project: designing technology for and with people who have aphasia. In *CUU '03*. 112–118.

[9] Medhi, I., Sagar, A. and Toyama, K. (2007). Text-free user interfaces for illiterate and semiliterate users. In *Information Technologies and International Development*. 4: 37–50.

[10] Meggs, P. B. and Purvis, A. W. (2005). *Meggs' History of Graphic Design*. Wiley.

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