

Cover Page

Title of submission: v4v: a View for the Viewer

Category of submission: Sketch

Name and full contact address (surface, fax, email) of the individual responsible for submitting and receiving inquiries about the submission: Mira Dontcheva, University of Washington, Dept. of Computer Science and Engineering, Box 352350, Seattle, WA 98195-2350, +1 206.543.5118 (voice), +1 206.543.2969 (fax)
mirad@cs.washington.edu

v4v: a View for the Viewer

Mira Dontcheva

University of Washington
mirad@cs.washington.edu

Michael F. Cohen

Microsoft Research
mcohen@microsoft.com

Steven M. Drucker

Microsoft Research
sdrucker@microsoft.com

Abstract

We present a **View for the Viewer** (v4v), a slide viewer that focuses on the needs of the viewer of a presentation instead of the presenter. Our design centers on representing the deck of slides as a stack embedded in a 3-D world. With only single button clicks, the viewer can quickly and easily navigate the deck of slides. We provide four types of annotation techniques and have designed a synchronization mechanism that makes it easy for the viewer to move in and out of sync with the presenter. We also supply alarms as a method for viewer notification. We evaluate our approach with a preliminary user study resulting in positive feedback about our design plus suggestions for improvements and extensions.

Keywords

Presentations, Note Taking, Slide Representation, User Interface Design, Visualization

Project/problem statement

We have all spent many hours sitting in presentations. Whether in large lecture halls or small meetings, presenting an idea in front of people is a primary means of communication. Ten years ago, a typical presentation involved a blackboard, transparencies, and an overhead projector. Today, presentations are given directly from laptops through a projector. Apart from the electronic format that enables some animation, there is little difference between the old style transparency presentations and those we see today.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Copyright © 2005 AIGA | The professional association for design.

Presentation software such as PowerPoint[10] typically has two modes; one for the author and one for the presenter. However, every presentation has a third party, the viewer, whose needs and priorities for the presentation are distinct from those of the author or presenter. While the author/presenter is interested in organizing content and conveying information clearly and concisely, a viewer is most often focused on absorbing information. As shown in previous work[6], the viewer must be able to take notes, stop and consider material they find confusing, and even inspect reference material. Furthermore, some viewers may only be interested in a portion of the presentation.

In this paper we present a new **View for the Viewer** (v4v), a viewer interface targeted at satisfying a number of the perceived needs of someone viewing a presentation (see margin note). The viewer we consider can be in a lecture hall in front of the presenter, in their office watching a live presentation on a video screen, or seeing the talk long after it is actually given. v4v combines a zoomable user interface[3] with a variety of annotation techniques, smooth animation, and simple synchronization mechanisms to create an intuitive and easy to use tool for viewing presentations. Note that in the design of v4v we purposely de-emphasized the needs of the presenter in order to better focus on the viewer's needs for organization, annotation, and exploration of the presented content. We first presented v4v in a pilot user study, which we use to gain further insights into how the system is perceived. In this paper, we assume that the viewer has a local copy of the presentation materials on their own device. This provides the basis for developing a viewer targeted interface.

v4v addresses the need to:

- quickly look through the deck of slides,
- explore the deck asynchronously from the presenter,
- remain aware of the location in the deck of the presenter,
- synchronize/desynchronize from the presenter at any time,
- annotate the slides while desynchronized,
- mark slides as important for later study,
- specify notifications for when the presenter has reached a certain section or slide

Background

A number of existing systems[1][4][6][7][12] focus on the classroom experience and how to facilitate presentations by enhancing student-presenter and student-student interactions. In one way or another, these systems provide most of the functionality necessary for v4v. Thus, our goal with this viewer is not to create a fully functional classroom experience but rather to explore a novel user interface for such a system that is designed specifically for the viewer. Our target environment is not so much a typical classroom setting but rather a conference presentation or business meeting where interaction with the presenter is not possible or as important as it is in class.

The projects that are most similar to our goal are StuPad[12] and Classroom Presenter[1]. StuPad was part of the eClass (formerly Classroom 2000) initiative and pioneered some of the work on integrating student information streams, such as personal notes, with presenter information streams, such as the slides, video, and audio. Classroom Presenter focuses on understanding digital ink: how it facilitates lecture environments and affects student-instructor interaction. Forum[6] was one of the first distributed presentation systems that explored a wide range of interaction mechanisms, such as voting polls, "speak" buttons, and messaging. Livenotes[7] and NotePals[4] focus on the interaction between viewers instead of the interaction between the presenter and viewer. In this work we address situations in which participation is minimal and focus our efforts on creating an intuitive browsing and exploration experience. MS OneNote[9] and Remarkable Texts[11] are systems designed specifically for note-taking and do not address synchronization with the presenter.

Challenge

Our design follows several constraints. First, we require no additional effort or material from the presenter, who must only share a copy of the slides with the audience at the beginning of the presentation. We also assume that the presenter will broadcast their actions, such as previous and next slide notifications, to the audience. Second, all interaction must be done with a single button click. We add this constraint to make v4v intuitive but also device independent so it can easily be used on tablets and PDAs. Last, we design for the live scenario when the viewer is interacting with the slides during the presentation. Our design, aside from the presenter synchronization mechanism, remains relevant in the offline scenario when the viewer explores a slide deck following a presentation.

Solution

A. Process

To create a fast and intuitive browsing experience we aim to provide a design that is compact and easily navigable. We first tried a 1D filmstrip representation, as it is the traditional format for slide decks but felt that as the number of slides increased the list quickly became unwieldy. Also, this representation made it difficult for the viewer to organize the slides without separating them from the list. A 2D grid representation is more effective with long presentations but breaks the linear structure of the deck. In contrast, a 3D stack representation maintains the slide order, allows for quick access to any slide, and also provides a mechanism for organizing the deck by splitting the stack into several parts, thus highlighting important slides (Fig 1). This splitting paradigm is similar to Mander et.al.'s[8] findings on human organization



Fig 1. The stack arrangement makes it easy to quickly browse through the slides. The user can further organize the deck into multiple stacks to highlight important slides.

behavior for information flow. Animation is used for slide transitions and view control because animation helps the user build mental maps of the content and maintain object constancy[2][5].

Since people use a variety of devices for notes, we include annotation mechanisms that are not only most common for note-taking during presentation but could be used on a variety of devices. We expect that viewers using a laptop will most often type their notes, while those using tablets will write their notes. We separate the different types of annotation according to the different tools used to create content. Although we can combine all the annotation techniques into one annotation canvas, we keep the mechanisms separate both for the simplicity of the user interface as well as the portability between devices.

We designed our presenter synchronization mechanism following previous work[1][12] but extended the idea of a synchronization button to the actual presenter window. Similar to Stupad and Classroom Presenter we use a color outline to aid the viewer in seeing whether the presenter is ahead or behind. We also give the viewer the ability to participate in only a part of the presentation with a notification mechanism, which visually alerts the viewer when the presenter has reached a certain slide.

B. Solution Details

1. Representation and Interaction

We represent the deck of slides as a front-to-back ordered stack of images. We allow the stack to split into parts but require the new substacks to preserve the original slide order. Time flows from top to bottom. The slides at the top of the screen are from the beginning of



Fig 2. To browse the slides (a), the viewer moves the mouse over the stack. The slide beneath the cursor is brought to the front. With just a click the stack splits and the focus slide is centered on the screen (b). The four tabs represent the different annotation paradigms (c).

the slide deck, while all slides below them are from subsequent parts of the deck. To browse through a stack of slides, the viewer moves the cursor over the deck of slides and the slide immediately under the cursor is scaled up and brought into view. To select a slide, the viewer clicks on the particular slide. The stack splits around this slide and the chosen slide is centered on the screen. The preceding slides appear in a stack above it and the subsequent slides appear in a stack below the current slide (Fig 2b).

Animation smoothly transitions from one configuration of slides to another. Smooth motion helps with perceived continuity and encodes information such as the direction of change. The 3D embedding of the 2D slides coupled with the animation provides an interface unique to our system. The viewer controls the camera view through the zoom parameter. To zoom out he clicks outside any slide. Conversely, to zoom into a particular slide, the viewer merely clicks on it. The viewer moves forward and backward through the deck by clicking on the corresponding down and up arrows on the left side of the interface.

2. Annotation

Viewing a presentation often involves taking notes and marking slides. We facilitate annotations with tabs. A tab is an icon for annotation content that is displayed when the pointer hovers over the tab or when the tab is clicked. Each slide includes four tabs that represent the different types of annotation mechanisms. Each tab is a different color and the color of the tab becomes more saturated when the tab includes content. The color difference allows the viewer to distinguish which slides have content and which do not. The four types of annotation mechanisms are ink (white), text (blue),

hyperlinks (green), and flagging (red). The white tab opens a semi-transparent canvas that covers the screen and allows the user to ink directly on the slide or in the space around the slide (Fig 2c). The blue tab opens a text box where the user can type. The green tab opens a web browser and contains the viewer's browsing history associated with that slide. The red tab is used to designate a slide as important. If a slide is flagged, it remains visible at all times and splits the stack into two new stacks, one with the slides preceding the flagged slide and one with the slides following. This type of annotation allows the viewer to easily organize the deck and highlight those slides that are of interest.

3. Presenter Synchronization

We show the presenter's current slide in a presenter window in the upper right corner of the screen. To synchronize with the presenter, the viewer clicks on the presenter window. To desynchronize the viewer clicks on any slide. To clarify the interface state (synchronized or not) we use the presenter window as a visual reminder. When the presenter's view is present, the viewer is not synchronized with the presenter; when the presenter's view is absent, the viewer's slides advance automatically with the presenter's.

A notification mechanism allows the viewer to explore the deck of slides freely without worrying about missing the part of the presentation that is most interesting. The viewer activates alarms that visually notify him when the presenter reaches a specific slide. The alarms are activated by clicking on the alarm clock icon present on each slide. When the presenter reaches a slide that has an activated alarm, the presenter space blinks to alert the viewer.

C. Results

We conducted a small user study to solicit feedback on v4v. Also, we wanted to compare v4v to traditional paper and pencil as well as Microsoft's OneNote, which is a popular note-taking package.

Experimental Design

Our pool of users included eight first and second year graduate students in computer science, who all attend lectures and presentations regularly and varied in their note-taking practices. The participants had no prior knowledge of OneNote or v4v. All of them reported that they use paper and pencil to take notes. They received minimal training prior to the commencement of each presentation.

Each study lasted 2 hours and was administered to 2 to 3 users simultaneously. The participants were shown portions of three pre-recorded videos of technical presentations. The presentations were typical of computer science conference presentations in that they were each presented by a different person, lasted 20 minutes, and involved a description of a problem, approach, and results. In the interest of time the participants were shown only a 10 minute representative segment of each presentation instead of the full talk. To simulate a live presentation, we broadcast previous and next messages to the participants, i.e. the audience.

In the complete 2-hour session, each user had a chance to use each of the note-taking paradigms. To encourage note-taking behavior, the participants were asked to answer 4-6 questions pertaining to the talk. To simulate typical presentation situations, we gave the participants half the questions before the presentation

and half the questions following the presentation. We split up the questions for two reasons: to give the participants an idea of the types of questions they would be asked; and to simulate a common scenario where a viewer may have some preexisting knowledge and have a specific interest in only part of the subject being presented.

Study Results

The overall response to v4v was positive but due to our small user pool there was little statistically significant difference between the ratings for the three interfaces. While there were many suggestions for improvements, the participants were excited to use v4v and explore its stack representation (see margin note). Most participants enjoyed exploring on their own and looking ahead in the deck of slides. They found the three dimensional structure of the presentation helpful for reviewing the presentation and their notes. The ability to quickly synchronize after jotting down notes was seen as a clear advantage over paper and MS OneNote. One complaint was the inability to see notes from previous slides as is possible on paper. Some participants requested that the canvas always be open and shown as active or inactive with transparency. All session participants agreed in their desire to combine text and drawing into one annotation type and add keyboard shortcuts. These requests are sensible for laptops but are not as appropriate for other devices. In the future, we plan to rethink our approach to annotation and specialization to particular devices. One interesting improvement was the desire to create paths through the presentation space. We can support multi-path slide decks by expanding the single stack paradigm to multiple stacks. Because slide order is conveyed with a top-to-bottom, front-to-back ordering,

User Feedback:

"But what I really did like about [v4v] was the automatic rejoin with the slides because you could just pause it, scribble down your notes, and then kind of hear what was going on in the background and then click and jump right back to the proper slide."

"I liked the stack based thing because you could group it into different sections."

"I wish I could have connected those two slides somehow, like put a reference back to this other one."

the horizontal axis of the screen remains unused and can be utilized for representing alternate paths. The extra screen space can also be used for manual organization of the slides. Restructuring a slide presentation, or repurposing a slide deck for new presentations, are both interesting avenues for future research. We also plan to rigorously evaluate the 3D slide representation and compare it with traditional 1D and 2D representations.

Conclusions

We present v4v, a **View for the Viewer**, a slide viewer that focuses on the needs of the viewer instead of the presenter. v4v provides a fast and easy navigation platform for a deck of slides and relies on intuitive synchronization mechanisms with the presenter. Our main contribution is the interface itself and its combination of navigation, annotation and synchronization mechanisms. We have taken first steps towards evaluating our design and plan to continue developing the system as we learn more about the viewer's needs.

Acknowledgements

We thank Krzysztof Gajos, Ratul Mahajan, and Wilmot Li for supplying materials for the user study. We also thank our user study participants for their time and feedback.

References

- [1] Anderson, R., Hoyer, C., and Wolfman, S., Anderson, R. "A study of digital ink in lecture presentation", In Proceedings of CHI'04, pp. 567-574.
- [2] Bederson, B., and Boltman, A. "Does animation help users build mental maps of spatial information?" In Proceedings of Infovis '99, pp. 28-35.
- [3] Bederson, B., Holland, J., Perlin, K., Meyer, J., Bacon, D., Furnas, G. "Pad++: a zoomable graphical sketchpad for exploring alternate interface physics", Journal of Visual Languages and Computing, 7(1):3-31, 1996.
- [4] Davis, R., Landay, J., Chen, V., Huang, J., Lee, R., Li, F., Lin, J., Morrey, C., Schleimer, B., Price, M., and Schilit, B. "Notepals: Lightweight note sharing by the group, for the group", In Proceedings of CHI'99, pp. 338-345.
- [5] Gonzalez, C. "Does animation in user interfaces improve decision making?" In Proceedings of CHI'96, pp. 27-34.
- [6] Isaacs, E., Morris, T., and Rodriguez, T. "A forum for supporting interactive presentations to distributed Audiences", In Proceedings of CSCW'94, pp.405-416.
- [7] Kam, M., Wang, J., Iles, A., Tse, E., Chiu, J., Glasner, D., Tarshish, O., and Canny, J. "Livenotes: a system for cooperative and augmented note-taking in Lectures", In Proceedings of CHI'05, pp. 531-540.
- [8] Mander, R., Salomon, G., and Wong Y., "A 'pile' metaphor for supporting casual organization of information", In Proceedings of CHI'92, pp. 627-634.
- [9] Microsoft. OneNote. <http://www.microsoft.com/office/onenote/>.
- [10] Microsoft. Powerpoint. <http://www.microsoft.com/office/powerpoint/>.
- [11] Remarkable Texts <http://graphics.cs.brown.edu/research/ReMarkableTexts/>
- [12] Truong, K., Abowd, G., and Brotherton, J. "Personalizing the capture of public experiences", In Proceedings of UIST'99, pp. 121-130.

Example Viewer Scenario

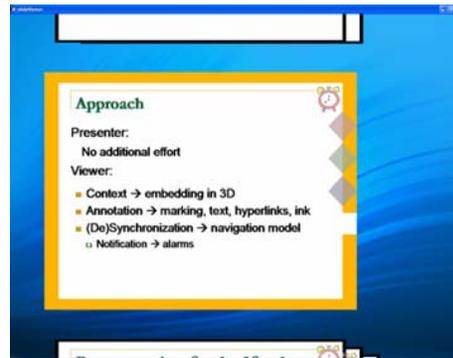


Fig 3a. Initially the viewer is synchronized with the presenter and his slides advance with the presenter's.

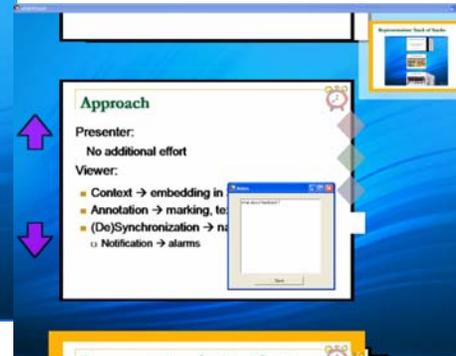


Fig 3b. The viewer can pause and write notes while the presenter advances to the next slide.

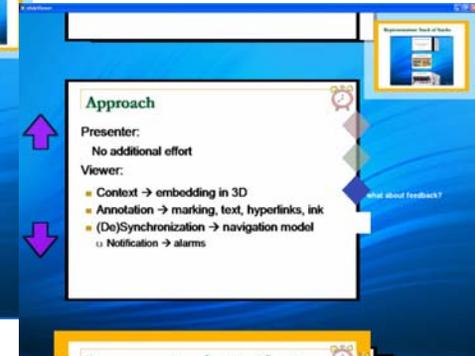


Fig 3c. The blue tab holds the notes written with the keyboard.

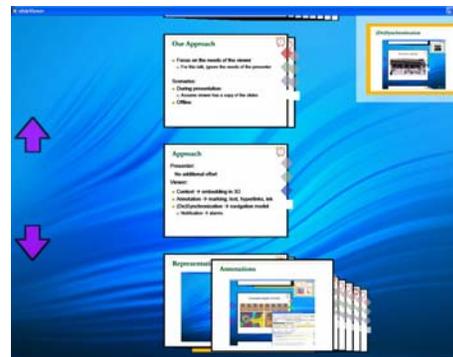


Fig 3d. The viewer zooms back and explores the subsequent slides.



Fig 3e. The white tab holds a canvas, which the user can use to mark the slide.



Fig 3f. The viewer zooms back again to look at another slide

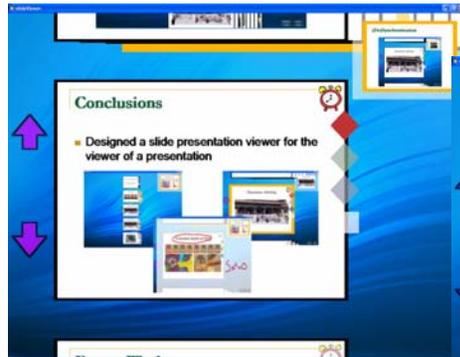


Fig 3g. The red tab flags this slide causing it to always remain visible. The viewer also sets the alarm clock, so he is notified when the presenter reaches this slide.

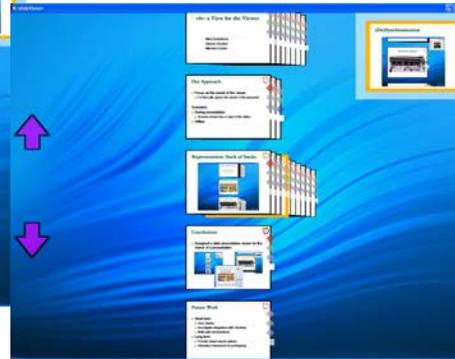


Fig 3h. The viewer continues to explore the deck, building up an organization through annotation.



Fig 3i. The presenter reaches the "Conclusions" slide and the presenter window blinks to notify the viewer.

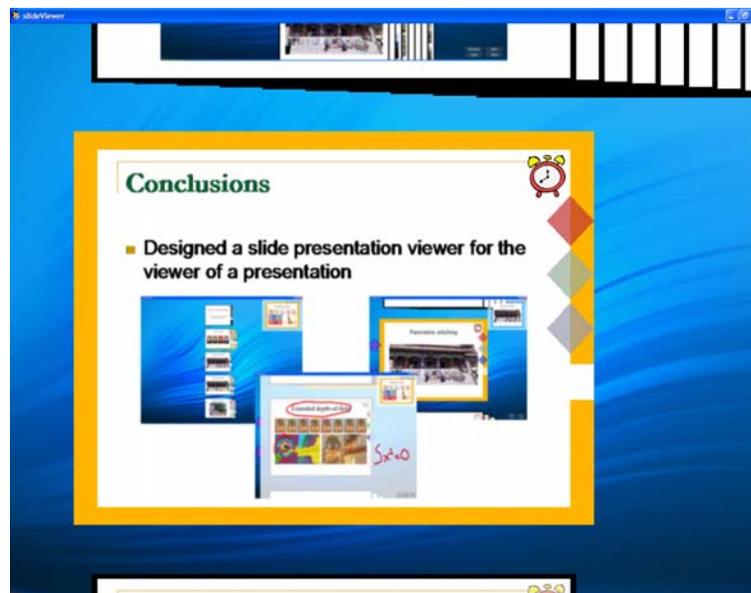


Fig 3j. The viewer clicks on the presenter window and is once again synchronized with the presenter.