A Slide-Ware Application to Support Discursive Presentations

Ryan P. Spicer Yu-Ru Lin Aisling Kelliher

Arts Media and Engineering Program, Arizona State University,
Tempe, AZ 85281 USA
{ryan.spicer, yu-ru.lin, aisling.kelliher}@asu.edu

1. ABSTRACT

Transdisciplinary collaborations call for dynamic, responsive slide-ware presentations beyond the linear structure afforded by traditional tools. The NextSlidePlease application addresses this through a novel authoring and presentation interface. The application also features an innovative algorithm to enhance presentation time management. The cross-platform Java application is currently being evaluated in a variety of real-world presentation contexts.

Categories and Subject Descriptors

H.5.2 [**Information Systems**]: User Interfaces; H.5.4 [**Information Systems**]: Hypertext/Hypermedia

General Terms

Design, Human Factors

Keywords

Slide-Ware, Presentations, Hypermedia, Hyperpresentations, Navigation

2. INTRODUCTION

Slide-ware presentations (such as those created using Microsoft PowerPoint, Apple KeyNote, and OpenOffice.org Impress) are ubiquitous in education, business, and government. In 2001, Microsoft estimated that over 30 million PowerPoint presentations were authored per day [1], and this number has no doubt increased in the intervening years. Slide-ware presentations are criticized, however, for a "rigid cognitive structure" which shapes presentations created through the applications [4]. Scholars argue that reliance on bullet-points and lack of supporting data can prevent presenters and their presentations from maximizing effectiveness in communication [2]. Traditional slide-ware applications meet many important communicative needs despite these potential pitfalls. Contemporary transdisciplinary working environments, however, require discussion and negotiation among diverse contributors. Traditional slide-ware does not afford this dynamic discssion; thus, a graph-based paradigm for presentation navigation is proposed. This paradigm permits the presenter to choose between many potential paths, allowing agile responses to audience interest or questions. In this paper, we describe a demonstration of NextSlidePlease, a novel slide-ware application that supports discursive presentations through graph-based navigation.

MM'09, October 19–24, 2009, Beijing, China.
ACM 978-1-60558-608-3/09/10.

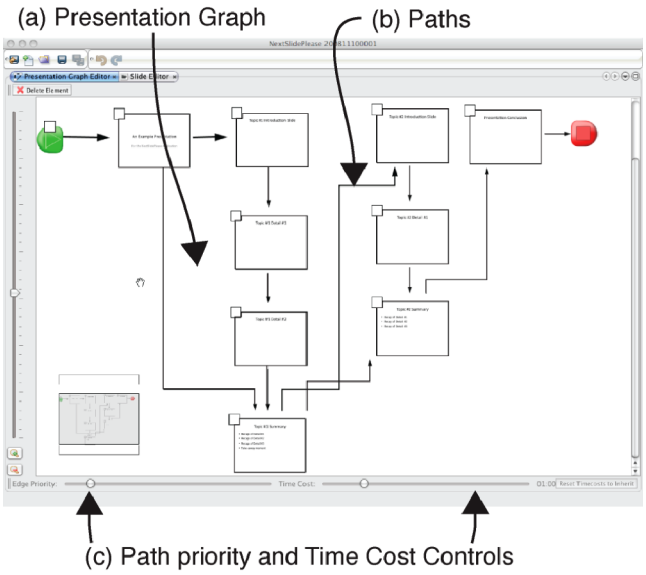

Figure 1. The NextSlidePlease Authoring Environment

3. THE APPLICATION

NextSlidePlease was first developed in response to research into the needs of business and academic users, as collected in a survey and semi-structured interviews [3]. Development is ongoing, driven by iterative user tests and evaluations as well as additional interviews and other input from expert slide-ware users.

NextSlidePlease addresses the common criticism of the fixed linear structure of slide-ware presentations by implementing a hypertext inspired navigation system. To handle the increased complexity created by such a fluid navigation system and to assist with the challenges of facilitating effective meetings in general, our application includes a novel time management and presentation path recommendation algorithm.

NextSlidePlease contains three primary components - an authoring environment, a presentation environment and a time management algorithm - which are described in detail below.

4. AUTHORING ENVIRONMENT

The primary activity in the authoring environment is the construction of a Presentation Graph (Figure 1a), which encodes the relationships among content in the presentation. In contrast to the 'film-strip' linear format familiar from traditional slide-ware applications, a NextSlidePlease presentation is stored and edited as a directed, weighted graph. In the authoring environment, the user imports slides stored in many image formats, created by traditional slide-ware tools or other image editors. These slides

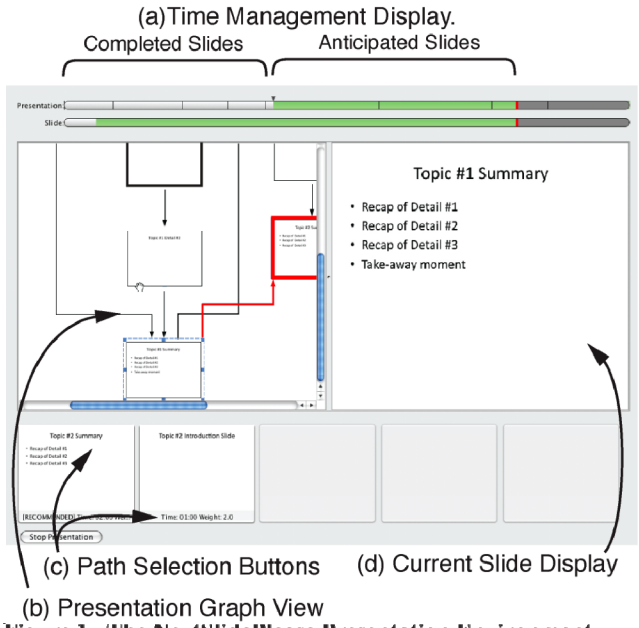

Figure 1. The NextSlidePlease Presentation Environment.

can be arranged on a 2D image plane, and paths, represented visually as graph edges (Figure 1b), can be created between them. Users may also set a Time Cost and Priority value for each path between slides using slider controls (Figure 1c). Priority describes the importance of following a given path, while time cost describes the predicted time to present the next slide. These values are used by the time management algorithm described in Section 2.3.

5. PRESENTATION ENVIRONMENT

The presentation environment allows the delivery of presentations created with NextSlidePlease. The environment requires two displays. One display, typically an LCD projector, presents the current slide to the audience. The second display, typically the laptop's screen, provides the presenter with feedback on the current state of the presentation (Figure 2). This information includes time-management, navigation choices, and the current slide. The time management display presents the remaining time for the current slide, the elapsed time, and the remaining time for the presentation (Figure 2a). The vertical lines demarcate transitions between slides.

The Presentation Graph view displays the presentation structure, (Figure 2b) with the currently-displayed slide outlined with a dashed border, and the suggested path highlighted in blue. The Current Slide Display (Figure 2d) mirrors the current slide.

During the presentation, the presenter navigates through the presentation by selecting among the possible paths leading from the current slide. In the interface, these choices are depicted as thumbnails of the slides, with the priority and the time cost of the edge displayed in text below (Figure 2c). The user may select the next slide using either the mouse or a keyboard command.

6. TIME MANAGEMENT ALGORITHM

The graph-based narrative paradigm frustrates traditional rehearsal and preparation techniques because the number of paths through a given set of slides exponentially increases as new edges are introduced. The novel time management/path-finding algorithm developed for NextSlidePlease assists the user in managing this cognitive load. The algorithm employs a linear constraint optimization technique to suggest an optimal tour from the current slide to the end of the presentation. This path is recomputed once per second, with respect to the presentation time constraints and current slide.

The algorithm permits two types of relaxation of the time constraint. The total time cost may be permitted to exceed the initial time constraint. In this case, the algorithm balances the increased priority of covering more slides against a penalty for increase in total time. Additionally, the algorithm can suggest that the user cover certain slides more rapidly in order to fit more slides into a given time. In this configuration, the algorithm balances added priority and decreased total time against a penalty for shortening slides.

7. DEMONSTRATION SETUP

In an ideal case, the demonstration hardware will include a laptop computer, an LCD projector, and a portable presentation screen. This equipment will be configured to emulate a traditional office conference room environment. If space for the demonstration is constrained, an LCD monitor positioned next to the laptop can be substituted for the projector and screen.

The laptop computer will run the NextSlidePlease application loaded with an example slide presentation. All data is stored locally, and no network access is required. The demonstration will allow participants to experience the primary authoring and presentation components of NextSlidePlease.

In the authoring environment, the participant can experiment with the presentation graph layout, creating multiple paths, assigning priority weights, and manipulating the time feature. In the presentation environment, the participant can use their own created layout or one of the provided layouts to gain an understanding of the NextSlidePlease presentation experience, experimenting with navigation and observing the consequences of content selection on time management feedback. In this way, the demo participant will experience all aspects of developing and delivering a discursive presentation using NextSlidePlease.

8. ACKNOWLEDGEMENTS

This work was generously supported by the National Science Foundation IGERT Grant No. 0504647.

9. REFERENCES

[1] I. Parker (2001). Absolute PowerPoint: Can a software package edit our thoughts. The New Yorker. 77: 76--87,

[2] G. Shaw, R. Brown and P. Bromiley (1996). Strategic stories: how 3M is rewriting business planning. Harvard Business Review 76(3): 41--49.

[3] R. P. Spicer and A. Kelliher (2009). NextSlidePlease: Improving Slideware User Interfaces for Dynamic Presentations, CHI '09: CHI '09 extended abstracts on Human factors in computing systems,

[4] E. R. Tufte (2006). Beautiful Evidence. Cheshire, Connecticut, Graphics Press LLC