Applying Programmable Browsing Semantics Within the Context of the World-Wide Web

Richard Furuta and Jin-Cheon Na
TEES Center for the Study of Digital Libraries
Texas A&M University
College Station, TX 77843-3112
{furuta,jincheon}@csdl.tamu.edu

ABSTRACT

We discuss application of caT (context-aware Trellis), which extends the Trellis Petri-net-based model of hypertext, towards specification of Web-browsable hypertexts that respond to factors that occur during their use. In addition to characteristics such as a reader's role (e.g., student, teacher, administrator, or parent) and the reader's browsing history, we also include factors that may not have been incorporated as directly before, such as measures of the external environment and the attributes/actions of other simultaneous readers. We use the term "responsive" hypertext to reflect the wide range of relevant factors.

Categories and Subject Descriptors

H.5.4 [**Information Interfaces and Presentation**]: Hypertext/Hypermedia; I.7.2 [**Document and Text Processing**]: Document Preparation—*Hypertext/hypermedia*

General Terms

Design, Theory

Keywords

caT, Trellis, Petri-net-based hypertext, context-aware hypertext, responsive hypertext

1. INTRODUCTION

As the reading audience for the World-Wide Web increases in number and diversity, and as the viewing venues broaden from wired workstations to wireless networks and miniaturized displays, so also does the need develop for Web-based hypertext collections that can be reshaped to respond to their user and use environments. In recent years, we have been examining the application of programmable browsing semantics, as introduced in the late 1980's by Furuta and Stotts' Trellis [3] and extended in caT, as a means for enabling a principled specification of such responsive hypertexts.

HT'02, June 11-15, 2002, College Park, Maryland, USA.

Trellis specifies hypertext applications with Petri nets. A key feature of the Petri net is that it combines a graph-based specification, automaton semantics, and a separate *marking* that defines the automaton's state. Trellis maps the Petri net to hypertext semantics, thereby defining not only the hypertext's nodes and links, but also its *browsing semantics*—where browsing semantics means the document behaviors (e.g., the sequence of information elements displayed and links made available) encountered during a reader's browsing session. Trellis browsing semantics are described as programmable, since the browsing behavior can be changed with localized change to the specification or marking.

2. RESPONSIVE HYPERTEXT

Over the past few years, we have extended the Trellis model to further support hypertext responsiveness. We call the revised model caT [2] (context-aware Trellis), reflecting its support for specifying characteristics of the external environment in which the hypertext is being used (e.g., physical location of the reader, time of day, user characteristics such as age and job title, etc.). Also, caT extensions permit the hypertext state to better encapsulate information about the reader as an individual, augmenting the less specific support for representing classes of readers provided in Trellis. The incorporation of a hierarchical net structure allows centralization of specifications implementing policies, while at the same time helping to localize the modifications needed if the policies are changed.

Trellis and caT implementations use a client-server architecture where the clients provide the user interface and the server enforces the model's semantics. Multiple clients can be active simultaneously, providing multiple views of a hypertext to a reader and/or allowing multiple readers to be browsing, and potentially interacting, over the hypertext.

We have implemented caT clients for use over the Web. Multiple content elements can be active at the same time, so caT's Web client creates a composite representation of the active content, directed by a template specification. This is then presented to the reader in a standard Web browser, and updated dynamically as the state of the net changes.

caT's extensions enhance the ability to specify hypertexts that respond to a wide range of factors. Factors that make it desirable to change the reading environment of a hypertext dynamically include those involved with a reader's characteristics or actions, other readers' characteristics or actions, features of the hardware environment, features of the real-world environment, and policy decisions made by the hypertext's managers. Response to factor changes is by modification of the hypertext's browsing behavior; i.e., a perceived change to the hypertext's browsing semantics. Factors and responses now will be discussed in more detail.

2.1 Causative Factors

Reader's characteristics: Hypertext customizations to meet a reader's individual characteristics may involve consideration of the reader's level of development, experience, training, skills, abilities, etc. As examples, an administrator's view of a hypertext may differ from that of a general reader; readers with handicaps may require different presentations of hypertext material.

Reader's actions: Changes in a hypertext's behavior based on the browsing history appeared quite early on in the context of literary hypertexts developed in Storyspace [1], where portions of a node's information might be visible only after other specific nodes had been visited. Key features in the browsing history include whether or not other specific areas of the hypertext have been visited, previous visits to the current area of the hypertext (e.g., has the current node been visited repeatedly or is this the first visit), and the length of the browsing history.

Interactions with other readers: Change may be based on the presence or absence of other readers or on the others' browsing history. For example, the presence or absence of help desk personnel may be reflected in the hypertext, a limited resource may make readers beyond a particular threshold number wait for access, a contest may only permit a fixed number of readers to gain access, or a quiz design may present each student with different sets of questions.

Hardware/networking environment: The differing capabilities of hardware devices (e.g., screen size, resolution, color) may suggest modifications to the hypertext's browsing semantics, as may differences in the available bandwidth or the cost of using that bandwidth.

Real-world environment: Time of day, geographical location, direction of physical orientation, altitude, proximity to physical objects or other readers, speed of travel, and other characteristics of the real-world environment may be reflected in a hypertext's browsing behavior.

External policies: Externally established administrative policies provide an additional source of potential factors; indeed a source that may appear quite arbitrary from the standpoint of the hypertext's specification. Access to a resource may be prohibited (or permitted only in abbreviated form) for classes of readers, from particular hardware environments, or from particular real-world environments (e.g., location, time of day).

2.2 Achieving Responsive Hypertext

In reaction to the factors just discussed, the browsing behavior of a responsive hypertext may be modified dynamically. Modifications may involve the hypertext's content, its link structure, and/or the hypertext's presentation. Perceived changes to the content may achieve several effects. Most directly, variants of the same content may be presented—for example, information in different languages, versions specialized for different experience backgrounds, or versions prepared for presentation on different media. Perceived link structural modifications may be modest in scale, achieving results similar to content changes, or may result in large-scale change to the hypertext's browsing behavior. A straightforward appearing modification is the addition or removal of portions of the structure (i.e., enabling or disabling links). For example, additional help links could be added on repeated visits to a portion of the hypertext. Modification of the presentation of information displayed to a reader during browsing presents a third category of potential responses to the causative factors. Content may be rendered for different devices (e.g., shown as printed text on displays or converted into synthesized speech for audio devices) and surface features of the formatting may be modified (e.g., fonts used in display).

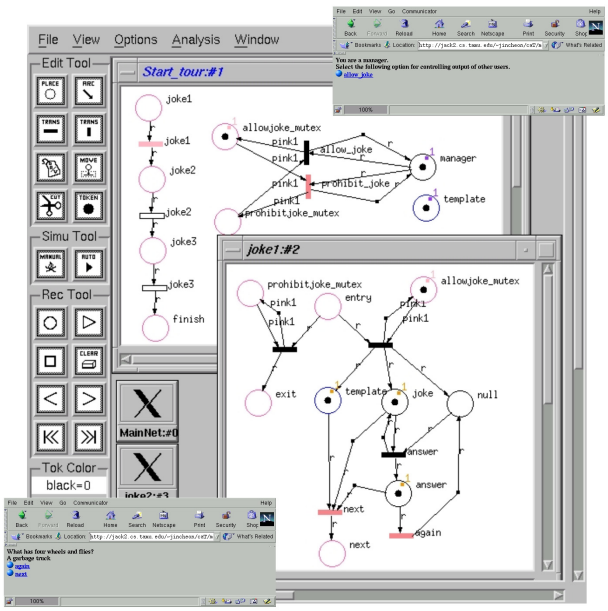

Figure 1: Example for interaction with other readers

3. EXAMPLE

We present a brief example to illustrate caT's responsiveness to changes in the factors (especially "Interactions with other readers"). Figure 1 shows the caT version of the χTrellis Petri net editor χTed with several subnets visible and two output displays. In the example, a user (i.e., a person with a role of a system manager) interactively affects the other users' displays, using the browser window shown in the upper right. In the state shown, readers can see a display of jokes, for example as in the browser window at lower left. The net display is shown in the larger window. If the manager decides to prohibit display of joke contents, selecting the `prohibit_joke` link (in the back window) results in addition of a token to access control place `prohibitjoke_mutex` and removal of a token from `allowjoke_mutex` in both of the visible subnets. The interested reader is referred to [2] for details of how these specifications are defined.

4. COMMENTS AND CONCLUSION

We continue to enhance the caT system to support hypertexts that are responsive to a wider range of factors. These specifications are becoming increasingly relevant as computing becomes more pervasive, and as the need increases for hypertexts that reflect the activities of individuals and their interactions with their community within the context of their physical and computational environment.

5. REFERENCES

[1] M. Joyce. Storyspace as a hypertext system for writers and readers of varying ability. In *Proceedings of Hypertext '91*, pages 381–387. ACM, Dec. 1991.

[2] J.-C. Na and R. Furuta. Dynamic documents: Authoring, browsing, and analysis using a high-level Petri net-based hypermedia system. In *Proc. ACM Symposium on Document Engineering (DocEng '01)*, pages 38–47. ACM, 2001.

[3] P. D. Stotts and R. Furuta. Petri-net-based hypertext: Document structure with browsing semantics. *ACM Transactions on Information Systems*, 7(1):3–29, Jan. 1989.