The Delivery of Multimedia Presentations in a Graphical User Interface Environment

Nathalie Colineau, Julien Phalip, Andrew Lampert
CSIRO – ICT Centre
Locked Bag 17, North Ryde
NSW 1670, Australia
phone: (+61) 2 9325 3100
{nathalie.colineau, julien.phalip, andrew.lampert}@csiro.au

ABSTRACT
A major issue in many domains is to present information to people that is tailored to their need, in such a way that it supports them in their tasks. In this paper, we present the Virtual Document Planner (VDP), a platform we developed for generating tailored interactive multimedia presentations in the surveillance domain. Integrated with the surveillance operators' graphical interface, the VDP provides tailored information delivery mechanisms that adapt the operators' information rich environment to their tasks and information needs.

Categories and Subject Descriptors
H.5.2 [**User Interfaces**]: Natural Language, Graphical User Interfaces; H.5.3 [**Group and Organization Interfaces**]: Synchronous Interaction; I.2.7 [**Natural Language Processing**]: Discourse, Language Generation.

General Terms
Design, Human Factors.

Keywords
Tailored information delivery, multimedia presentation generation, discourse approach, task-sensitive user interface.

1. INTRODUCTION
In the context of innovative Airborne Early Warning and Control (AEW&C) platforms, we have investigated the delivery of multimedia information to support the surveillance operators in their tasks. In their current environment, operators access information that comes from heterogeneous sources, and that is delivered in a number of ways, across several displays.

To support operators in their tasks, we have enhanced their graphical interface with information delivery mechanisms that tailor this information rich and complex environment to their tasks and information needs. These delivery mechanisms have been integrated within the operators' interface, minimizing the numbers of displays and ensuring the most relevant information is always available and prominently displayed. To this end, we have implemented a platform, called the Virtual Document Planner (VDP), for generating tailored interactive multimedia presentations. Based on an analysis of the operators' tasks and an understanding of the context in which operators perform their tasks (see [2]), the VDP determines the relevant information to present to the operators. This information is then organized and delivered on the operators' graphical interface.

The VDP is based on a typical Natural Language Generation (NLG) architecture, where the linguistic resources are separate from the engine, and, like many multimedia presentation systems (e.g., [1], [4] and [6]), it employs a discourse approach based on Rhetorical Structure Theory (RST) [5]. However, our approach is different in a number of ways. Unlike these systems, the presentation we produce requires a real integration between the information tailored by the VDP and the information already on screen. Indeed, the aim was not to generate the operators' interface, essentially a radar display, but rather, to augment their radar display with additional information. To do this, we have designed and integrated a workspace to the operators' radar display. This is where most of the information generated by the VDP is displayed. When it is needed, the information can be also directly overlaid on the radar display, like, for example, the visualization of track's kinematics information or the display of flight routes. In addition, while we use simple techniques to do the media allocation (not as sophisticated as [8]), we have focused our effort on two specific points that we present in this article:

- Developing a flexible approach to specify the layout of the presentation. The design is based on a template approach which allows us to keep a clear separation between our delivery platform and the delivery device; and
- Developing a new approach to resolve the media objects' synchronization. The originality consists of using the underlying discourse structure of the presentation to link the media objects together and determine their behavior.

2. THE SURVEILLANCE DOMAIN
Surveillance operators work with information that comes from various sources and that may be visualized on different displays. The primary source of information is real-time information from sensors such as radars that indicate the location of airborne objects in the area. In addition to the sensors' returns, other information may come from a variety of sources. It is the role of the surveillance operators to collect and integrate this additional information to facilitate the interpretation of the battle space. The combined output of the surveillance operators' work is the *Recognized Air Picture*. To facilitate the operators' task, it was proposed to provide their interface with mechanisms that select and aggregate relevant information from a variety of sources, and

IUI'06, January 26–February 1, 2006, Sydney, Australia.
ACM 1-59593-287-9/06/0001.

then, deliver it into one integrated display rather than distributed across several screens.

3. PLATFORM OVERVIEW

To effectively deliver information on the operators' interface, it was necessary to adapt their current environment centered on the radar display. The aim was not to design the next generation of their interface, but to propose a solution on how the additional information could be delivered to the operators, and how the operators would interact with it. Based on a study of their current interfaces and work practices (internal technical report), we proposed to add to the radar display an integrated workspace that gathers and delivers information tailored by the VDP, as illustrated in Figure 1. The objective was simple: minimize visual clutter and only display task- and situation-tailored information. This integrated workspace was also a way to overcome the potential intrusiveness of unsolicited delivery of information.

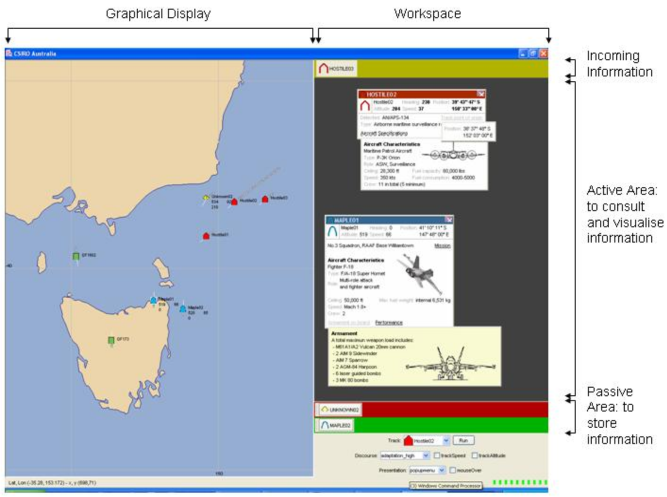

Figure 1: Graphical User Interface Organization

Figure 2 provides an overview of the platform architecture. The platform is composed of five components:

Simulator: The primary input to the GUI is from a simulator module, which generates tracks (i.e., simulated series of radar returns, which are associated with a single individual aircraft).

Data Repository: The GUI uses a series of data stores to record and maintain information about objects within a simulation (e.g., tracks).

Graphical User Interface (GUI): The GUI is the interface through which operators interact with the system. It is a prototype, which implements a subset of features that allow us to demonstrate the technology without seeking absolute real word verisimilitude. Our GUI is implemented in Java. The display of spatial information is handled using the OpenMap toolkit (http://openmap.bbn.com).

Application Programming Interface (API): The API operates as an interface layer between the VDP and the GUI. Its role is to interpret information sent by the VDP that will be rendered by the GUI. To this end, we use BeanShell (http://www.beanshell.org), a free Java source interpreter. By using the BeanShell interpreter, we create a very flexible scripting interface for the GUI.

Virtual Document Planner (VDP): The VDP generates multimedia information tailored to operators' tasks. It is based on a typical Natural Language Generation (NLG) architecture, where the linguistic resources are separate from the engine. The presentations generated are scripting commands that are interpreted by the API, and then, rendered by the GUI. The VDP has been used in different applications to generate a variety of documents.

4. GENERATING A PRESENTATION

The VDP operates in three stages as illustrated in Figure 2: a) the planning process; b) the assembly process, and c) the realization, which consists of producing a script of the presentation that is then sent to the API for interpretation and rendered by the GUI. In the following, we briefly outline our approach to assemble and realize the presentation.

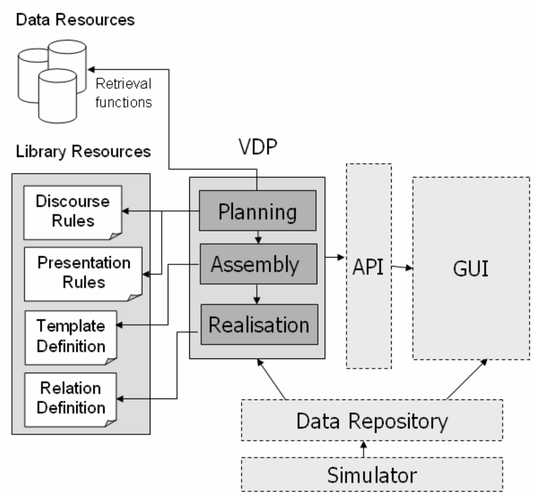

Figure 2: Overview of the generation process

The first step when generating a presentation of information is to decide what information is to be included, and how to convey this information. In the VDP, this takes place in two planning stages: 1) the content planning, in which discourse rules are selected and combined to specify what content should be included (i.e., what needs to be conveyed) and how to organize it coherently; 2) the presentation planning, in which presentation rules are selected and combined to specify the most appropriate way to convey the content selected. This is also during the presentation planning that the content is actually retrieved from the data resources.

4.1 The Assembly Process

Once the presentation has been planned, the next step consists of assembling the presentation, and, in particular, specifying how the content is to be laid out. In this application, the content consists mostly of figures, labels and short specifications. It is mainly presented in windows and delivered on a GUI. To lay out the content in window and control precisely the every details of the window layout, we use a template approach. The template representation used by the VDP is different from other approaches as it uses a generic representation that it is not expressed in any specific programming language. The templates are simple declarative XML files, and combined together, they specify how a particular presentation layout has to be achieved. To realize the actual layout, the VDP templates are then associated with XSLT style sheets (XSLT is an extensible style sheet Language for defining XML document transformation and presentation), which describe how to display content in a syntax understood by the delivery device (in our case the GUI environment). In doing so, we ensure a clear separation between the VDP and the rendering device.

The assembly is a bottom-up process. It starts at the leaves of the discourse tree built during the planning process, selects the templates to be used, and step by step embeds the templates together specifying how each piece of information must be laid out. When the VDP templates are associated with the corresponding style sheets, this also automatically specifies how the style sheets are to be combined.

While assembling the templates, the assembly process annotates further the discourse tree structure. The annotations consist of indicating for each node what is the media object built. These annotations are used later on by the realization process when interpreting the discourse relations holding between the nodes (i.e., discourse segments). The final step for the VDP is the realization of an output script that contains all the presentation specifications. It is then interpreted by the API and eventually rendered on the GUI.

4.2 The Realization Process

The realization process is responsible for generating an output script embedding all the information necessary to render the presentation on the display. This is done during a second bottom-up pass through the discourse tree. At this point, the annotations made during the assembly process are interpreted and the discourse relations holding between nodes are processed. To ensure a coherent interaction process with the user, it is important to coordinate the behavior of the media objects constituting the presentation. An easy way to determine how to link them together is to use the discourse tree structure created during the planning process. Our approach consists of reasoning about the discourse structure underlying the presentation to derive the relationship between the media objects, and then use the type of discourse relation that links a media object to another in the discourse tree structure to assign them a particular behavior.

There exist various languages to specify the temporal dimension of interactive multimedia presentations (e.g., [3] [7]) enabling very detailed specifications of the media object synchronization. However, our approach to assigning a behavior to media objects is different in a number of ways:

- First, in coordinating the media objects' behavior based on the underlying discourse structure, it provides a motivation as why and how media objects should be synchronized. The discourse structure provides us with an understanding of how information is related together, and how each piece of information is contributing to the whole;
- Second, it provides an easy way to define or change objects' behavior by modifying the association between the discourse relations and the behavior rules, or by changing the discourse relation holding between two nodes;
- Finally, it ensures consistency in the way all the media objects behave across the presentation. Indeed, two objects linked by the same discourse relation will share and also exhibit the same behavior.

Once the output script has been generated, the file is interpreted by the API. At this stage, the behavior rules that control the media object synchronization are activated. They ensure that media objects of similar importance behave in a similar way. In doing so, this approach enables the VDP to deliver not just content, but also to specify interaction and synchronization constraints.

5. CONCLUSION

We have presented the VDP platform designed for the delivery of multimedia information. This platform has been integrated within a surveillance environment to augment the operators' graphical interface with information delivery mechanisms.

6. ACKNOWLEDGMENTS

We wish to thank Robert Tot for his work on the task analysis and modeling and the RAAF surveillance operators and fighter controllers who participated in the task analysis interviews. We wish to thank also Cécile Paris for participating in discussions and for her comments on an early version of this paper. We acknowledge the support of Boeing (Research Agreement WT-SIDA 010.2) and CSIRO.

7. REFERENCES

[1] André, E. and Rist, T. (1995). Generating coherent presentations employing textual and Visual material. In *Artificial Intelligent Review,* Special volume on the Integration of Natural Language and Vision Processing, 9 (2/3). 147-165.

[2] Colineau, N. and Paris, C. (2003). Task-Driven Information Presentation. In *Proc. of OZCHI'03*, Brisbane, Australia, Nov 25-28.

[3] Dalal, M., Feiner, S., McKeown, K., Zhou, M., Hollere, T., Shaw. J., Feng, Y. and Fromer, J. (1996). Negotiation for automated generation of temporal multimedia presentations, In *ACM Multimedia Conference*, p. 55-64, Boston, MA. USA.

[4] De Carolis, B., De Rosis, F., Andreoli, C., Cavallo, V. and De Cicco, M.L. (1998). The dynamic generation of hypertext presentations of medical guidelines. In *The New Review of Hypermedia and Mult*imedia, 4, 67-88.

[5] Mann, W. and Thompson S. (1998). Rhetorical Structure Theory: Towards a Functional Theory of Text Organization. In *Text*, 8(3): 243-281, 1988.

[6] Rutledge, R., Bailey, B., van Ossenbruggen, J., Hardman, L. and Geurts, J. (2000). Generating Presentation Constraints from Rhetorical Structure. In *Proc. of the 11th ACM Conference on Hypertext and Hypermedia*, San Antonio, Texas, May 30 – June 3, 19-28.

[7] SMIL (Synchronized Multimedia Integration Language) http://www.w3.org/TR/2005/REC-SMIL2-20050107/

[8] Zhou, M., Wen, Z. and Aggarwal, V. (2005). A graph-matching approach to dynamic media allocation in intelligent multimedia interfaces. In *Proc. of IUI'05*, January 9-12, San Diego, California, US, pp.114-121.