A Unified Framework for Constructing Multimodal Experiments and Applications

Adam Cheyer[1], Luc Julia[1] and Jean-Claude Martin[2]

[1] SRI International, 333 Ravenswood Avenue, Menlo Park, CA 94025 USA
cheyer@ai.sri.com, julia@speech.sri.com
[2] LIMSI-CNRS, BP 133, 91403 Orsay Cedex, France
martin@limsi.fr

Abstract. In 1994, inspired by a Wizard of Oz (WOZ) simulation experiment, we developed a working prototype of a system that enables users to interact with a map display through synergistic combinations of pen and voice. To address many of the issues raised by multimodal fusion, our implementation employed a distributed multi-agent framework to coordinate parallel competition and cooperation among processing components. Since then, the agent-based infrastructure has been enhanced with a collaboration technology, creating a framework in which multiple humans and automated agents can naturally interact within the same graphical workspace.

Our current endeavor is the leveraging of this architecture to create a unified implementation framework for simultaneously developing both WOZ simulated systems and their fully automated counterparts. Bootstrapping effects made possible by such an approach are illustrated by a hybrid WOZ experiment currently under way in our laboratory: as a naive subject draws, writes, and speaks requests to an interactive map, a hidden Wizard responds as efficiently as possible using either standard graphical interface devices or multimodal combinations of pen and voice. The input choices made by both subject and Wizard are invaluable, and the data collected from each can be applied directly to evaluating and improving the automated part of the system.

Keywords: Multimodal user interface, Wizard of Oz user studies, distributed agents

1 Introduction

Wizard of Oz (WOZ) simulations have proven an effective technique for discovering how users would interact with systems that are beyond the current state of the art [Oviatt 96,Oviatt 97]. However, WOZ systems are costly to build from scratch and are rarely reusable across domains. Furthermore, it is often difficult to evaluate how lessons learned from the experiment directly impact the design and effectiveness of a real application.

In this paper, we will first describe a fully automated prototype (presented at CMC95 in Cheyer and Julia 95) that was inspired by a multimodal WOZ simulation [Oviatt 96], and then explain how its functionality evolved as the application was used. We then describe how the fully automated system was enhanced to serve as a hybrid WOZ simulation where the actions of both the naive subject and the expert Wizard are objects of the experiment. The approach is put forth as a general-purpose, unified framework for simultaneously constructing multimodal WOZ experiments and their fully functional versions, such that the two synergistically improve each other.

2 A Fully Automated Multimodal Map Application

2.1 Description

Our multimodal map application provides an interactive interface on which the user may draw, write, or speak. In a travel planning domain (Figure 1), available information includes data

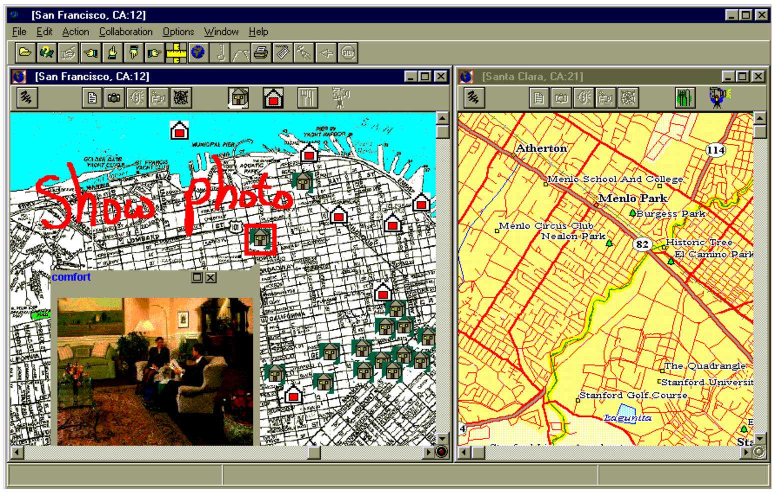

Fig. 1. A Multimodal Map Application

about hotels, restaurants, and tourist sites that have been retrieved by distributed software agents from commercial Internet World Wide Web sites. The types of user interactions and multimodal issues handled by the application can be illustrated by a brief scenario featuring working examples taken from the current system.

Sara is planning a business trip to San Francisco, but would like to schedule some activities for the weekend while she is there. She turns on her laptop PC, executes a map application, and selects San Francisco.

```
Ex1.1    [Speaking] Where is downtown?
         Map scrolls to appropriate area.
Ex1.2    [Speaking and drawing region] Show me all hotels near here.
         Icons representing hotels appear.
Ex1.3    [Writes on a hotel] Info?
         A textual description (price, attributes, etc.) appears.
Ex1.4    [Speaking] I only want hotels with a pool.
         Some hotels disappear.
Ex1.5    [Draws a crossout on a hotel that is too close to a highway]
         Hotel disappears
Ex1.6    [Speaking and circling] Show me a photo of this hotel.
         Photo appears.
Ex1.7    [Points to another hotel]
         Photo appears.
Ex1.8    [Speaking] Price of the other hotel?
         Price appears for previous hotel.
Ex1.9    [Speaking and drawing an arrow] Scroll down.
         Display adjusted.
Ex1.10   [Speaking and drawing an arrow toward a hotel]
         What is the distance from this hotel to Fisherman's wharf?
         Distance displayed.
Ex1.11   [Pointing to another place and speaking] And the distance to here?
         Distance displayed.
```

Sara decides she could use some human advice. She picks up the phone, calls

Bob, her travel agent, and writes Start collaboration to synchronize his display with hers. At this point, both are presented with an identical map, and the input and actions of one will be remotely seen by the other.

Ex2.1 [Sara speaks and circles two hotels]
Bob, I'm trying to choose between these two hotels. Any opinions?
Ex2.2 [Bob draws an arrow, speaks and points]
Well, this area is really nice to visit. You can walk there from this hotel.
Map scrolls to indicated area. Hotel selected.
Ex2.3 [Sara speaks] Do you think I should visit Alcatraz?
Ex2.4 [Bob speaks] Map, show video of Alcatraz.
Video appears.
Ex2.5 [Bob speaks] Yes, Alcatraz is a lot of fun.

For this system, the main research focus is on how to generate the most appropriate interpretation for the incoming streams of multimodal input. Our approach employs an agent-based framework to coordinate competition and cooperation among distributed information sources, working in parallel to resolve the ambiguities arising at every level of the interpretation process:

- low-level processing of the data stream: Pen input may be interpreted as a gesture (e.g., Ex1.5: crossout, Ex1.9: arrow) by one algorithm, or as handwriting by a separate recognition process (e.g., Ex1.3: info?). Multiple hypotheses may be returned by a modality recognition component.
- anaphora resolution: When resolving anaphoric references, separate information sources may contribute to resolving the reference:
  - Context by object type: For an utterance such as show photo of the hotel, the natural language component can return a list of the last hotels talked about.
  - Deictic: In combination with a spoken utterance like show photo of this hotel, pointing, circling, or arrow gestures might indicate the desired object (e.g., Ex1.7). Deictic references may occur before, during, or after an accompanying verbal command.
  - Visual context: Given the request display photo of the hotel, the user interface agent might determine that only one hotel is currently visible on the map, and therefore this might be the desired reference object.
  - Database queries: Information from a database agent can be combined with results from other resolution strategies. Examples are show me a photo of the hotel in Menlo Park and Ex1.2.
  - Discourse analysis: Discourse can provide a source of information for phrases such as No, the other one (or Ex1.8).

  This list is by no means exhaustive. Examples of other resolution methods include spatial reasoning (the hotel between Fisherman's Wharf and Lombard Street) and user preferences (near my favorite restaurant).
- cross-modality influences: When multiple modalities are used together, one modality may reinforce or disambiguate the interpretation of another. For instance, the interpretation of an arrow gesture may vary when accompanied by different verbal commands (e.g., scroll left vs. show info about this hotel). In the latter example, the system must take into account how accurately and unambiguously an arrow selects a single hotel.
- addressee: With the addition of collaboration technology, humans and automated agents all share the same workspace. A pen doodle or a spoken utterance may be meant for either another human, the system (Ex2.1), or both (Ex2.2).

A first version of this prototype system was presented at CMC95; the system has evolved since then in several ways. First, the user interface was redesigned with an eye toward practicality (Figure 1). Whereas the design for the user interface of the original system was patterned directly after that of the WOZ experiments, which for obvious reasons encourages the user to produce strictly pen/voice input, the redesign provides standard GUI devices (e.g., scrollbars, toolbars, menus, dialog boxes) if that is the most efficient means of expressing the intent. The human-human collaboration mode is new. The map interface has also been augmented to accommodate multiple windows, each representing a workspace with a separate context (e.g., city, viewport position, zoom factor, shared vs. private space) The distributed multimodal interpretation process, as described above, has evolved considerably, particularly with respect to cross-modality ambiguity resolution. Finally, the multimodal map has been applied to a applications outside of the travel planning domain [Moran et al. 97].

2.2 Implementation

The map application is implemented within a multiagent framework called the Open Agent Architecture (OAA) [3]. The OAA provides a general-purpose infrastructure for constructing systems composed of multiple software agents written in different programming languages and running on different platforms. Similar in spirit to distributed object frameworks such as OMG's CORBA or Microsoft's DCOM, agent interactions are more flexible and adaptable than the tightly bound object method calls provided by these architectures, and are able to exploit parallelism and dynamic execution of complex goals. Instead of preprogrammed single method calls to known object services, an agent can express its requests in terms of a high-level logical description of what it wants done, along with optional constraints specifying how the task should be performed. This specification request is processed by one or more Facilitator agents, which plan, execute and monitor the coordination of the subtasks required to accomplish the end goal [Cohen et al. 94].

The core services of the OAA are implemented by an agent library working closely with a Facilitator agent; together, they are responsible for domain-independent coordination and routing of information and services. These basic services can be classified into three areas: agent communication and cooperation, distributed data services, and trigger management. For details on these topics and information about how to build applications using the OAA, refer to [DMartin et al. 98].

The map application is composed of 10 or more distributed agents that handle database access, speech recognition (Nuance Communications Toolkit or IBM's VoiceType), handwriting (by CIC) and gesture (in-house algorithms) recognition, natural language interpretation, and so forth. As mentioned in the previous section, these agents compete and cooperate to interpret the streams of input media being generated by the user. More detailed information regarding agent interactions for the multimodal map application and the strategies used for modality merging can be found in [Cheyer and Julia 95,Julia and Cheyer 97].

In addition to the system described in this paper, the OAA has been used to construct more than 20 different applications, integrating various technologies in many domains: multirobot control and coordination [Guzzoni et al. 97], office automation and unified messaging [Cohen et al. 94], front ends [Julia et al. 97] and back ends [DMartin et al. 97] for the Web, and development tools [DMartin et al. 96] for creating and assembling new agents within the OAA. Other agent-base multimodal applications are described in [Cheyer 98,Moran et al. 97, Moore et al. 97].

[3] Open Agent Architecture and OAA are trademarks of SRI International. Other brand names and product names herein are trademarks and registered trademarks of their respective holders.

3 A Hybrid Wizard of Oz System

For any WOZ experiment, the runtime environment must generally provide the following facilities:

1. An interface, for the subject, which will accept user input (without processing it), transmit the input to a hidden Wizard, and then display the results returned by the Wizard.
2. An interface, for the Wizard, which provides a means for viewing the subject's input, and for rapidly taking appropriate action to control the subject's display.
3. Automated logging and playback of sessions to facilitate the data analysis process.

The multimodal map application already possesses two qualities that help the fully automated application function part of a WOZ experiment: the system allows multiple users to share a common workspace in which the input and results of one user may be seen by all members of the session – this will enable the Wizard to see the subject's requests and remotely control the display; the user interface can be configured on a per-user basis to include more or fewer GUI controls – the Wizard can lay out all GUI command options, and still work on the map by using pen and voice (Figure 2). Conversely, the subject will be presented with a map-only display.

To extend the fully automated map application to be suitable for conducting WOZ simulations, we added only three features: a mode to disable the interpretation of input for the subject, domain-independent logging and playback functions that leverage the agent collaboration services, and a separate message agent for sending WOZ-specific instructions (e.g., Please be more specific.) to the user with text-to-speech and graphics.

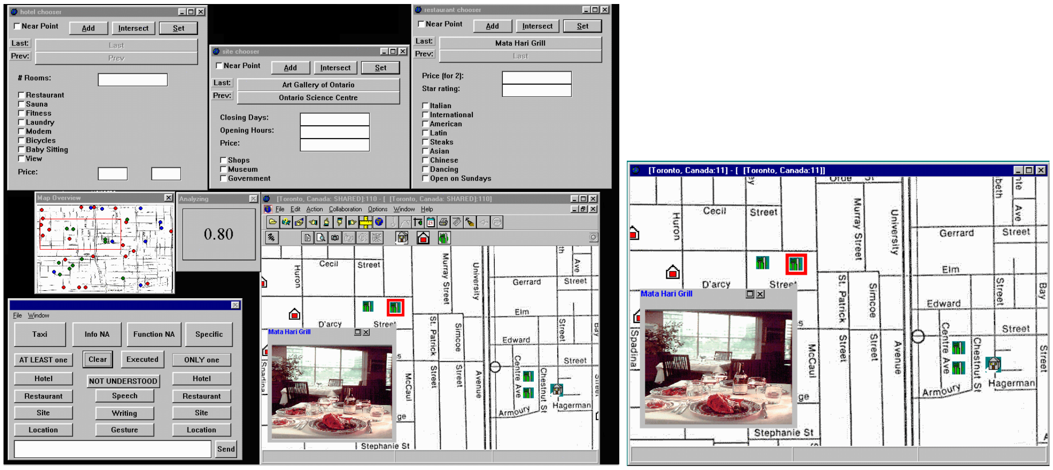

Fig. 2. The Wizard interface and the Subject interface

The result is a hybrid WOZ experiment: While a naive user is free to write, draw, or speak to a map application without constraints imposed by specific recognition technologies, the hidden Wizard must respond as quickly and accurately as possible using any means at his or her disposal. In certain situations, a scrollbar or dialog box might provide the fastest response, whereas in others, some combination of pen and voice may be the most efficient way of accomplishing the task. In a single experiment, we simultaneously collect data input from both an unconstrained new user (unknowingly) operating a simulated system – providing answers about how pen and voice are combined in the most natural way possible – and from an expert user (under duress) making full use of our best automated system, which clarifies how well the real system performs and lets us make comparisons between the roles of a standard GUI and a multimodal interface. We expect that this data will prove invaluable from an experimental standpoint, and that since all interactions are logged electronically, both sets of data can be directly applied to evaluating and improving the automated processing. How well did the real system perform for the Wizard? How well would the fully automated system have fared on the actual data produced by the new user if there were no Wizard? Are there improvements that could be made to the speech grammar, modality merging process, or other aspects of the system that would significantly increase overall performance? How much do the changes actually improve the system?

Performing such experiments and evaluations in a framework where a WOZ simulation and its corresponding fully functional end-user system are tightly intertwined produces a bootstrap effect: as the automated system is improved to better handle the corpus of subject interactions, the Wizard's task is made easier and more efficient for future WOZ experiments. The methodology promotes an incremental way of designing an application, testing the design through semiautomated user studies, gradually developing the automated processing to implement appropriate behavior for input collected from subjects, and then testing the finished product while simultaneously designing and collecting data on future functionality – all within one unified implementation. The system can also be used without a Wizard, to log data about how real users make use of the finished product.

4 Conclusions and Future Work

We have described a framework and a novel approach for simultaneously developing a WOZ simulation and a working prototype for multimodal applications. This integration encourages bootstrap effects: data and results obtained from the user experiment can directly improve the automated processing components, making the Wizard's responses more efficient. The architecture is generic, allowing an application/experiment developer to freely select programming languages, input and output modalities, third-party recognition engines, and modality combination technologies (e.g., neural nets, slot-based approaches, temporal fusion).

We are currently in the process of applying the framework described in this paper to conduct a data collection effort, of approximately 30 subjects, that focuses on spatial references in multimodal map-based tasks. The data will be analyzed along a several dimensions by using TYCOON, a theoretical framework for evaluating multimodal user studies, as described in [JCMartin et al. 98]. Results from these experiments for both the WOZ subjects and the Wizard using the automated system will be made available in future publications.

In addition, we are working on a version of the Open Agent Architecture for public distribution on the Web. The libraries and tools will include enough functionality for researchers to begin experimenting with constructing systems of the style described here. Information regarding distribution availability and terms will be made available on the OAA homepage (http://www.ai.sri.com/∼oaa).

5 Acknowledgments

This paper was supported in part by National Science Foundation/Advanced Research Project Agency Grant IRI-9619126. We would like to thank Andy Kehler, Jerry Hobbs, and John Bear for valuable discussions and comments on earlier drafts. Thanks also to Wayne Chambliss for his excellent Wizardry.

References

Cheyer, A. MVIEWS: Multimodal tools for the video analyst. Conference on Intelligent User Interfaces (IUI'98), San Francisco, January 1998.

Cheyer A., Julia L. Multimodal maps: An agent-based approach. In Proceedings of CMC95, Amsterdam, 1995. 103-113.

Cohen, P., Cheyer, A., Wang, M., Baeg, S. An Open Agent Architecture. In AAAI Spring Symposium. Stanford University, March 1994. 1-8.

Guzzoni, D., Cheyer, A., Julia, L., and Konolige, K. Many robots make short work. AI Magazine, Vol. 18, No. 1, Spring 1997. 55-64.

Julia, L., and Cheyer, A. Speech: a privileged modality. In Proceedings of EUROSPEECH'97, Rhodes, Greece, September 1997.

Julia, L., Cheyer, A., Neumeyer, L., Dowding, J., and Charafeddine, M. HTTP://WWW.SPEECH.SRI.COM/DEMOS/ATIS. In AAAI Spring Symposium, Stanford University, March 1997. 72-76.

Martin, D., Cheyer, A., and Moran, D. Building Distributed Software Systems with the Open Agent Architecture. To appear in a forthcoming agent conference. Also available online at http://www.ai.sri.com/~oaa + "Publications", 1998.

Martin, D., Cheyer, A., and Lee, GL. Agent development tools for the Open Agent Architecture. In Proceedings of the International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology. London, April 1996.

Martin, D., Oohama, H., Moran, D., and Cheyer, A. Information brokering in an agent architecture. In Proceedings of the Second International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology. London, April 1997.

Martin, J.C., Julia, L., Cheyer, A. A theoretical framework for multimodal user studies. CMC98. 1998.

Moore, R., Dowding, J., Bratt, H, Gawron, JM, Cheyer, A. CommandTalk: A spoken-language interface for battlefield simulations. Fifth Conference on Applied Natural Language Processing, Washington, D.C., April 1997.

Moran, D., Cheyer, A., Julia, L., and Park, S. Multimodal user interfaces in the Open Agent Architecture. In Proceedings of IUI-97. Orlando, Jan. 1997. 61-68.

Oviatt, S. Multimodal interfaces for dynamic interactive maps. Proceedings of CHI96, April 13-18, 1996. 95-102.

Oviatt, S., De Angeli, A., Kuhn, K. Integration and synchronization of input modes during multimodal human-computer interaction. Proceedings of the workshop "Referring Phenomena in a Multimedia Context and their Computational Treatment", ACL/EACL'97, July 11, 1997, Madrid. 1-13. http://www.dfki.uni-sb.de/imedia/workshops/mm-references.html.