

InfoWiz: An Animated Voice Interactive Information System

Adam Cheyer

SRI International
Artificial Intelligence Center
333 Ravenswood Ave.
Menlo Park, CA 94025 - USA
cheyer@ai.sri.com

Luc Julia

SRI International
STAR Laboratory
333 Ravenswood Ave.
Menlo Park, CA 94025 – USA
julia@speech.sri.com

The InfoWiz Kiosk

The InfoWiz project is centered around the idea of putting an interactive kiosk into the lobby of SRI. People who have a few minutes to spend will be able to learn something about the institute, enjoy themselves, and walk away with a good feeling of having seen something unusual and interesting.

One of the design decisions of the project is to use speech recognition as the primary mode of input to the system. We feel that if we are successful in building an effective spoken language interface, this should contribute to the user's overall impression of ease of use, advanced technology, and general intelligence of the system. If we are not successful, we can always put back a touch screen.

To encourage spoken interactions, we added an animated character, a cartoon wizard, who attempts to engage the user in conversations about SRI. As Don Nielson (to whom this system is dedicated) used to say, talking to a computer without an animated persona who can speak back is like talking to a wall. Initially, we made our own animations based on a popular clip-art figure. In later incarnations of the interface, we adopted Microsoft Agent's Merlin character to provide the graphics for the system.

Interactions with the InfoWiz

After an initial audio and visual welcome from the InfoWiz, the user is presented with a screen containing a web browser and the InfoWiz himself (Figure 1). The wizard's job is to instruct, advise, suggest, and demonstrate; he can lead tours of the information space, help the user navigate the complete set of SRI's web pages, and he can answer questions that come up along the way. The InfoWiz is able to cross window boundaries and interact with any part of the display, flying and pointing to various places on the screen.

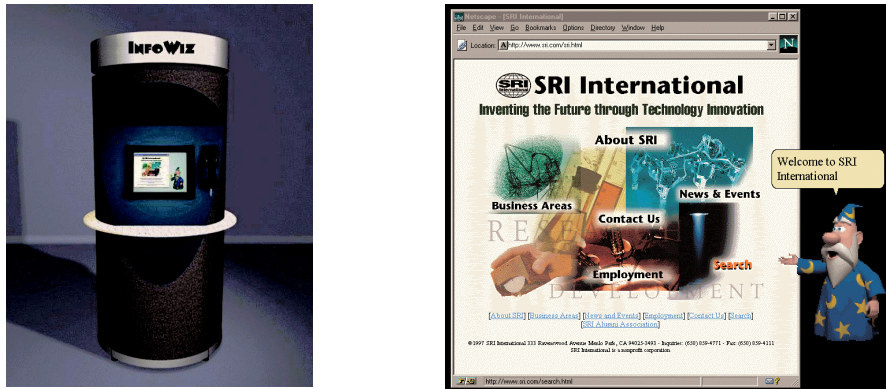


Figure 1. The InfoWiz Kiosk and a Sample Screen

All audio interactions with the system occur through a telephone next to the kiosk display, chosen because a telephone provides privacy as well as a familiar interface. Users can make natural spoken requests about information presented on the current web page or about general topics related to SRI. The Wizard is capable of answering questions and providing supplementary information (e.g., videos) about the InfoSpace¹. When the user is looking at web pages outside of its own domain (the InfoSpace), the InfoWiz can scroll pages or help navigate links, but of course, does not have any special supplementary information about these pages.

Implementation Approach

The system is implemented on top of the Open Agent ArchitectureTM (OAA[®]) [5], a distributed infrastructure that provides the means for bringing together multiple component technologies in a flexible, plug-and-play manner. Components can be written in different programming languages² and be distributed over multiple computers.

Several technologies are integrated using the OAA to create the InfoWiz Kiosk. The basic elements, as pictured in Figure 2, are character animation graphics, speech recognition, natural language interpretation, dialog management, text-to-speech generation, and a repository of knowledge about the InfoSpace.

¹ InfoSpace refers to the set of pages in a web site for which there is coded knowledge accessible to InfoWiz.

² Current languages include Java, C, C++, Visual Basic, Prolog, Lisp, Delphi, and WebL.

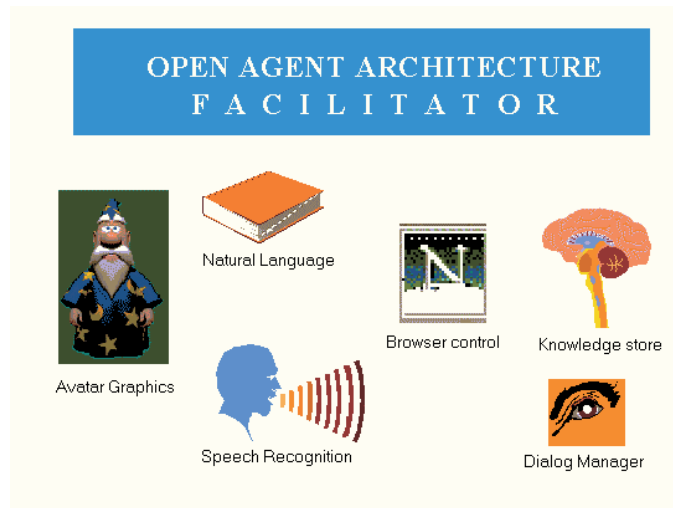


Figure 2. The InfoWiz Architecture

The plug-and-play nature of OAA has allowed us to experiment over time with various versions and vendors for the component technologies. For speech recognition, we have tried IBM's ViaVoice, and are now using an SRI-developed speaker-independent continuous speech recognition system commercialized in a spin-off called Nuance Communications³. For the web browser, we started with an agent-enabled controller for NCSA's Mosaic on Unix, and now interchangeably use Netscape's Navigator or Microsoft's Internet Explorer under Windows. Natural language (NL) processing is handled by a mixture of SRI's DCG-NL parser and Nuance's NL API. Wizard animation was initially handled using a Java-based animation package called the Gamelet Toolkit⁴. We later incorporated an early version of DFKI's PPP Animated Presentation Agent [1], and are currently using the Microsoft Agent character animation tool⁵.

The Illusion of «Intelligence»

The biggest challenge of our project is to create a system that appears intelligent, easy to use, and not overly constraining to an untrained, not necessarily technically aware person who literally walks in off the street. As illustrated by many avatar-based applications, creating intelligence in an animated persona often depends on carefully selecting a domain and then

³ <http://www.nuance.com>

⁴ <http://www.hipbone.com/GameletToolkit>

⁵ <http://www.microsoft.com/msagent>

matching interface presentation techniques that maximize the user's appreciation of «appropriateness» for responses generated by the character. In James Lester's educational application PlantWorld [4], the possible user interactions are constrained enough to script responses for all combinations of user input. With Microsoft Office's animated assistants, Bayesian reasoning is used to select responses from the set of pre-existing help pages, and multiple answers are presented to the user, increasing likelihood that a correct response is contained in one of the responses [3]. In systems developed by Barbara Hayes-Roth's Virtual Theater group and commercialized by Extempo⁶, characters have a crafted personality and hobbies outside of the target domain, providing a more human-like flavor to responses [6].

Several unique characteristics of the InfoWiz Kiosk domain can be exploited when designing the system. The primary goals of our kiosk are to entertain, to impress, and to educate, in that order of importance. One fortuitous result is that we can often substitute humor for accuracy and come out ahead. In other words, if the wizard can't say something intelligent, it often works *in this domain* to say something funny. Another unique property is that the system is intended for visitors who will most likely only ever use the system once or twice, for a total duration of several minutes. Whereas humor works well as a fallback for short durations, keeping humor fresh over a long period is a hard problem, and even funny dialog can become annoying as part of longer, more goal-driven interactions.

The InfoWiz's appearance of «intelligence» is based on the following:

- Knowledge Repository (KR): Knowledge is semiautomatically constructed that covers the information presented in a collection of web pages. Knowledge elements are distributed into five categories for which the knowledge engineer populates a set of scripts and definitions: topics of interest that the user might want more information about (e.g., robots, history); high-level questions a user might ask (e.g., «Who is the CEO of SRI?» or «What can I say?»); acronym definitions (e.g., «What does SRI stand for?»); information to display or actions to perform when a user arrives at a specific web page; and system-specific definitions such as unusual pronunciations for vocabulary words (used by speech recognition and text-to-speech).
- Dialog Management: Scripts control the InfoWiz's movements and utterances, but the Dialog Manager can adapt them to reflect context.

⁶ <http://www.extempo.com>

Provided services include reference resolution (e.g., «Ok, show it to me now.» or «No, the other one.»), question-answering, interaction history (don't provide the same information twice), goal management (e.g., «Make sure the user sees all information on Artificial Intelligence.»).

- User Feedback: The system always finds a way to indicate what was understood. If a user asks about «SRI», a misrecognition error could provoke «Tide detergent? I know I have information about that somewhere...»
- Misrecognition Strategies: When the speech recognition completely rejects the user's utterance as not understandable, the InfoWiz employs a calculated combination of stalling (e.g., «I'm sorry, what were you saying?»), instruction (e.g., «Try asking about high-level topics such as History or Research.»), and after several recognition failures, mixes in proactive guidance (e.g., «Hey! I'd like to take you to see some of my favorite places.»). The system detects how well it is performing and adapts its behavior over time.

Populating the Knowledge Repository

The most important component of InfoWiz's «intelligence» is the knowledge repository – it's the KR that enables InfoWiz to recognize topics and questions a user might have in a given domain and produce appropriate responses. If the InfoWiz project is going to succeed over time, we will need tools and methods for facilitating the population of the KR, for guaranteeing the KR's consistency over time as the target web pages change, and for updating appropriate data files for other components in the InfoWiz system (e.g., speech recognition, text-to-speech generation).

A primary concept used by the KR is the topic. In the first phase of our current knowledge population process, a developer runs an information extraction program based on SRI's FASTUS technology [2] over the target set of web pages to produce a list of all names, organizations, and complex noun phrases (terms) contained within. Using this list, the developer groups elements in the list into topics. For example, the terms «AIC», «AI Center», and «Artificial Intelligence Center» may all be ways of referring to the same organization. The combined topic «AICenter» will contain all the ways the topic was talked about in the text (useful for speech recognition) and the addresses of the pages containing references to the topic (potential places to bring the user to if the user asks about the topic). The developer then continues entering knowledge about the extracted topics by using a knowledge editor, called IWEdit (Figure 3).

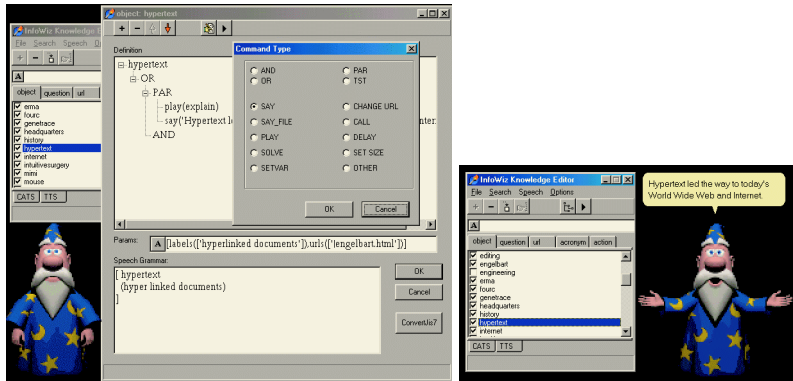


Figure 3. The InfoWiz Knowledge Editor (IWEdit)

IWEdit presents a graphical interface for creating and manipulating knowledge stored remotely by an OAA database agent. Knowledge scripts are expressed in a simple language that can be represented as a tree of operators (e.g., sequential-AND, parallel-AND, random-CHOOSE-ONE, test-CONDITION) and statements that influence InfoWiz and web browser behavior, perform computations, manipulate variables, and interact with other InfoWiz components (e.g., the dialog manager) using OAA's interagent communication language. Scripts are attached to knowledge elements (e.g., topics, questions, web pages, definitions) and are triggered under control of the dialog manager based on system- and user-level events.

As knowledge scripts are entered for each topic and knowledge element, IWEdit provides visual feedback about which elements have not yet been defined, helping ensure consistency of the knowledge base with respect to the web content. Once the knowledge population task is complete, a developer can generate grammars and dictionary files used by speech recognition or text-to-speech components of the system. IWEdit can also be used to test the system interactively, executing scripts and subdialogs, and a developer can use IWEdit to tweak a running InfoWiz system by modifying knowledge on the fly.

Status and Future Work

The InfoWiz Kiosk is currently installed in a public demonstration space where a fair number of people, both from within SRI and external visitors, can use the system. All interactions are logged, and to date we have performed three iterations of updating speech grammars and knowledge scripts according to what we have learned from the user data. The system has attained a level of

coverage that is approaching good enough to place in SRI's main lobby, enabling the InfoWiz Kiosk to serve as a showcase of SRI technology.

InfoWiz has already been adapted to work as an automated guide that can answer questions about PowerPoint presentations. We are also adapting the InfoWiz to work in an Internet or E-commerce domain: the wizard lives on the user's machine and learns preferences about that user. When he or she navigates to a site enabled with an InfoWiz knowledge server, the InfoWiz wakes up and downloads a knowledge template, thereby becoming an expert on the site, and can then interactively help guide the user to find the desired information.

References

- [1] Elisabeth Andre, Thomas Rist, and Jochen Muller. Guiding the User Through Dynamically Generated Hypermedia Presentations with a Life-Like Character. In *Proceedings of International Conference on Intelligent User Interfaces (IUI-98)*. San Francisco, January 1998.
- [2] J. Hobbs, D. Appelt, J. Bear, D. Israel, M. Kameyama, M. Stickel, and M. Tyson. FASTUS: A Cascaded Finite-state Transducer for Extracting Information from Natural-language Text, in *Finite State Devices for Natural Language Processing* (E. Roche and Y. Schabes, eds.), Cambridge MA: MIT Press, 1996.
- [3] E. Horvitz, J. Breese, D. Heckerman, D. Hovel, and K. Rommelse. The Lumiere Project: Bayesian User Modeling for Inferring the Goals and Needs of Software Users, in *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, Madison, WI. pp 256-265. July 1998.
- [4] James Lester, Patrick FitzGerald, and Brian Stone. The Pedagogical Design Studio: Exploiting Artifact-Based Task Models for Constructivist Learning. In *Proceedings of International Conference on Intelligent User Interfaces (IUI-97)*, Orlando. January 1997.
- [5] David Martin, Adam Cheyer, and Douglas Moran. The Open Agent Architecture: A Framework for Building Distributed Software Systems. *Applied Artificial Intelligence: An International Journal*, vol. 13, pp. 91--128, January-March 1999.
- [6] D. Rousseau and B. Hayes-Roth. *Personality in Synthetic Agents*, Knowledge Systems Laboratory Report No. KSL 96-21, Stanford University, 1996.