



UNIVERSITY OF
CALGARY

**A tutorial report for SENG 609.22
Agent Based Software Engineering
Course Instructor: Dr. Behrouz H. Far**

Tutorial on Conversational Agents

By
Steve Daeninck

1.0 Introduction

People are very social animals. Communication is extremely important to us since we use it everyday in all aspects of our lives. We learn through communication, pass on ideas and accomplish tasks we cannot do alone. People use a variety of methods to communicate. The methods most used are speech and writing, which convey ideas through language. Lesser methods, but still important ones are hand movements and facial features, which are more useful in describing emotion when used in conjunction with speech.

Technology is supposed to make our lives easier, but communicating with technology can be difficult sometimes. To use a computer, the commands must be known first. Sometimes you must know the correct question to ask before the required answer can be given. With all of this, it can be difficult to find the information we require.

Conversational agents are relatively new. Their purpose is to facilitate communication between people and technology, or just to assist people in finding what they need. To do this they interact with a person in natural language. Conversational agents use conversation and in some cases movement and facial features to convey information. The person does not have to learn special commands or operations to use a system. On top of that, the conversational agent will be more natural to use because it is more comfortable to use as it emulates normal human interaction.

2.0 What are Conversational Agents

Conversational agents are software systems that emulate people to convey information to a user in a natural way. There are different types of conversational agents. Some use speech or text, and some, called embodied; use body movements along with facial features and language to convey information to the user. They use natural language and are usually limited to a small domain of information. The use of natural language makes it easier for people to get what they want out of the agent. Questions like “What is ...”, “How do I ...”, “Where can I find ...” can be used. The agent will respond in similar language, even offering information that was not asked for but is deemed relevant to the topic the user is discussing.

3.0 Components of a Conversational Agent

There are certain components to a conversational agent that are common for all. Each Agent must have the following components to function:

1. A form of input is required. This could be text entry, speech recognition or a combination of both. Text input is the most common and easiest to implement. Speech recognition is much harder. Good speech recognition software is customized to the user. There are far more errors if it is not.
2. The agent must have a natural language parser. This is used to interpret what the user has said. The parser turns the language of the user into string/format it can easily understand.
3. Some form of decision-making process to respond to the user. This process can take a variety of forms, the first being standard and the remaining three more experimental learning approaches, but with greater promise:
 - A standard and common approach is using string matching to analyze the user input. The agent finds common strings and words from the database that match those from the user’s input. The response is generated using string templates. Then whenever certain words or strings occur, the template is used to output a response.
 - Case Based Reasoning (CBA) is a learning approach. In this approach the agent retrieves and adapts a previous solution and applies it to the problem¹. If there is no previous case, then the problem of responding to

the user input is done with rule-based decision making. When an answer has been given, the response is stored in the database. If the response was satisfactory for the user, then it is stored as successful. If the response was not satisfactory, then it is stored as a failure. Case Based Reasoning uses both the failures and the successes to determine appropriate responses.

- Ripple Down Rules (RDR) is another learning approach that generates responses². This is a mechanism for incrementally enhancing the scripts of conversational agents. The system has a basic script starting point for responses. The user can change or correct any responses from the agent over time. The RDR system will record these changes as exceptions for the particular user. This way, from the same basic starting point, many different responses can be generated from the same input, but from different users.
 - Batch Learning by Induction Over Examples is the last approach. Logs of human interactions are stored and common patterns in those interactions are sought. These patterns are used to construct a set of rules or decision tree for generating responses for the user. Eventually these rules will describe the user-preferred behavior of the system.
4. The conversational agent must be able to convert its response generated in the previous step into natural language for the user. This will be a reverse of the natural language parser.

Those components described above are the minimum necessary for a conversational agent. Others can be added to make the agent more human like. Embodied agents are those that have virtual bodies and faces. By using facial expressions and body language, the agents can convey more information and seem more like a real person to the user. This will put the user more at ease when using the agent, which is the goal of conversational agents.

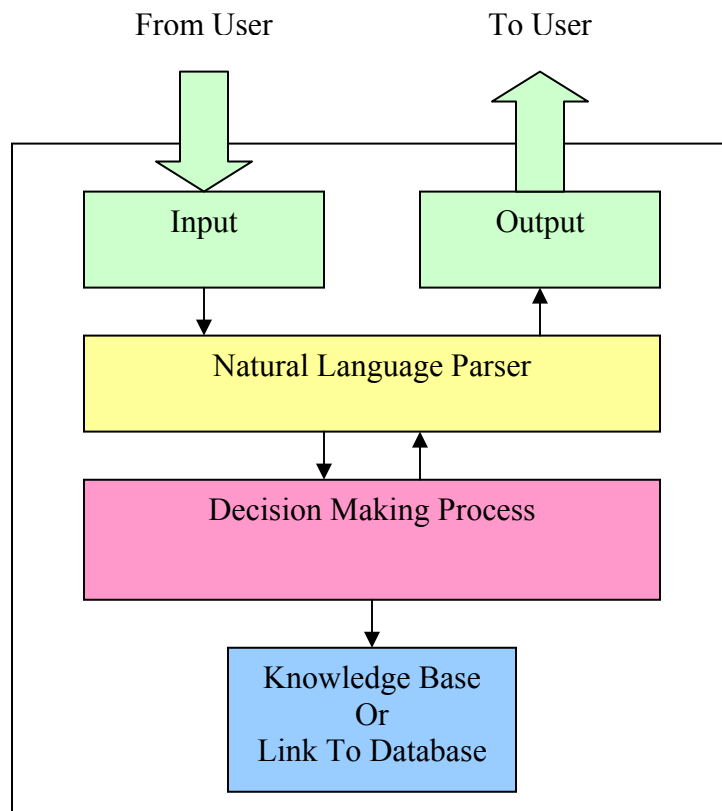


Figure 1. Conversational Agent Components

4.0 Uses for Conversational Agents

There are many uses for a conversational agent. They could be used for:

- Replacing the touch-tone phone automated services. Instead of having to listen to multiple menus of “Press 1 for...”, “Press 2 for ...”, etc, a conversational agent could be used. So the person could speak to the agent as if it were a person and get the information needed without listening to a lot of menu options.
- Tutoring people on any kind of topic from school to computer application use.
- As web guides. Finding information for people or the sites that they want to visit.
- Automatically answering Frequently Asked Questions. This could be implemented for any business or system.
- Web Based Company Representative. When visiting a company web page, the agent could answer any questions that might come up about any aspect of the company from stock price to projects.

- As a virtual support agent to answer less demanding IT questions. The agent could represent an actual member of a company's IT support staff. The real support person would only get involved if the agent could not answer the question. This would free up the actual support worker to concentrate on difficult problems.

This list is far from complete. These agents could be used for any task that requires user interaction with non-unique requirements.

5.0 Agent Based Design

Conversational Agents are natural candidates for agent based system design. The conversational agent needs to communicate, it needs to have access to a database or use its own knowledge base, most need some form of mobility and some learn as they are used. Combinations of these 4 requirements qualify the conversational agent as a software agent.

Each agent would require the basic modules described in section 3.0, a communication protocol, and an interface with the user.

The communication protocol it uses would depend on the system type and where it is used. An Internet application would require the communication protocol be able to talk to other agents and access service on the Internet. A telephone application would require only communication with the user and possibly other agents if it were a multi-agent system.

The agent would need to have access to all the information in its conversation domain. This would be accomplished either by having the information entirely in its own knowledge base, or having access to a database. The database would be more appropriate for agents that are customized for users. The agent could have the custom information in its knowledge base and use that to gather the responses from a central database. Some of the agents would learn as they interact with people. The learned material would have to be stored inside the agent's own knowledge base.

The interface with the user would vary according to the application. In some cases text would be used, so a GUI would be appropriate. In the case of speech, the interface would be transparent to the user.

These conversational agents could also be used as a user interface for any other agent based system. It would need to use the same communication protocol as the system, or use a facilitator agent to communicate. Other than that it would only need to change its data or knowledge base so it could perform the actions required. In Open Agent Architectures, where the system can be easily extended, then adding a conversational agent as the user interface would require minimal effort. The ability for humans to be

able to communicate easily and naturally with the community of agents is an important design requirement of the OAA³.

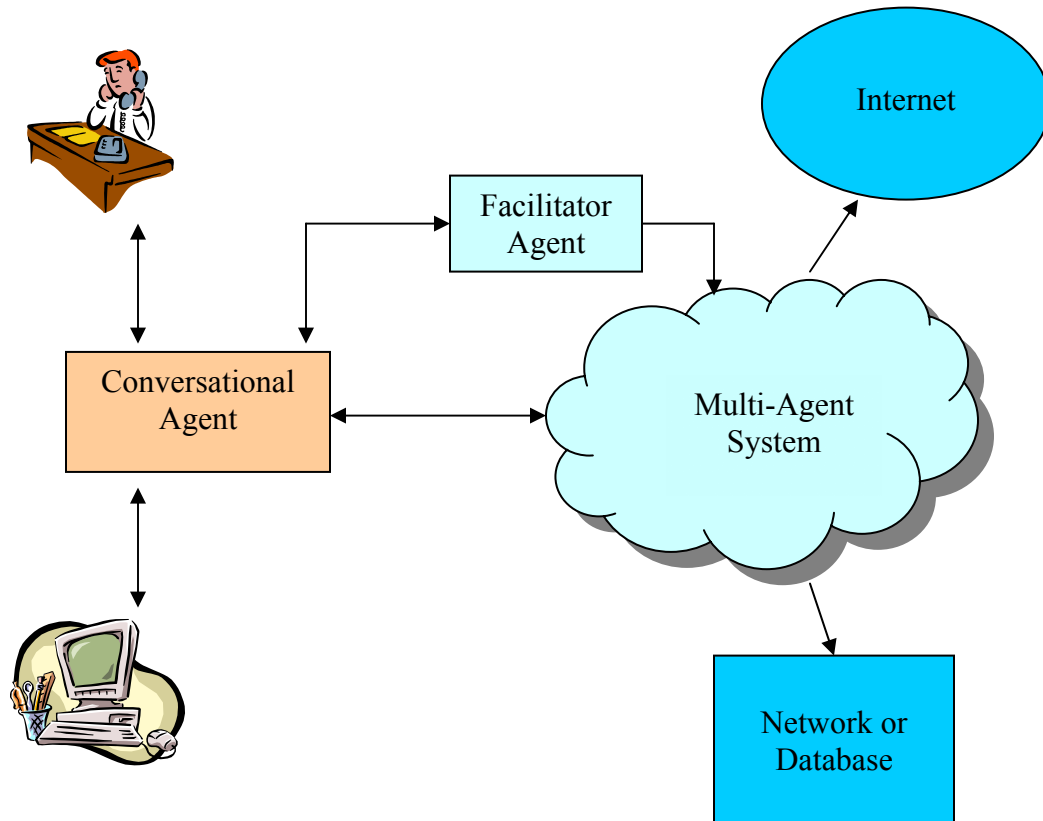


Figure 2. Conversational Agents As User Interface

6.0 Technology

The technology to use would depend on the application. There would be many choices to use for Internet applications. There are a lot of technologies out to enable easy use of Internet services. Examples from Dr. Far's Travel Agency System Handout include:

- Universal Description, Discovery and Integration (UDDI) with Web Service Description Language (WSDL). These are the equivalent of the yellow pages.
- Resource Description Framework (RDF), which is a sign mounting approach.
- JINI, a Java based system, in which each service has a proxy object that the client interacts with to use the service. Services advertise themselves to the lookup service. This is used for web and network based systems.

For non-internet systems, the best choice would be Java because it has a large amount of available API's and it is easily portable. Each system will need to run on the Java Virtual Machine though.

7.0 Conclusion

Conversational agents are not a revolutionary idea or technology. They simply take something like a FAQ database and make it more useable by people. The goal is to make a system that responds to people naturally, in the same way that they speak to each other. Instead of asking the system a question and getting an answer, the user has a conversation with the system to get the desired information. The needs of a Conversational agent are such that the best way to design them is to use agent-based software. Perhaps in the future, people will no longer be able to tell if they are speaking with a software agent or a person, unless they are face to face.

References

1. David B. Leake. Combining Rules and Cases to Learn Case Adaptation. 1995. <http://www.ai-cbr.org/theindex.html>
2. Australian Knowledge Acquisition Workshop. From Multiple Classification RDR to Configuration RDR. 1997. <http://www.cse.unsw.edu.au/>
3. Dr. B.H. Far. SENG 609.22 Session 4: Agent-based System Architecture and Organization. 2002. <http://www.enel.ucalgary.ca/People/far/>
4. Dr. B.H. Far. SENG 609.22 Travel Agency System. 2002. <http://www.enel.ucalgary.ca/People/far/>
5. Joseph P. Bigus, Jennifer Bigus. Constructing Intelligent Agents Using Java Second Edition. New York, 2001.
6. MIT Media Laboratory. <http://www.media.mit.edu>