A Multi-Agent Control and Automation Architecture with Integrated Flexible User Communication Agents

Ahmed Kamel
Offutt School of Business
Concordia College
Moorhead, Minnesota, USA
Email: kamel@cord.edu

***Abstract*—In this paper, we present a flexible integrated multi-agent automation and control architecture. The architecture relies on a knowledge-based module. This module is custom designed to every application and is thus open to personalization to other domains requiring a similar control scheme. It, thus, provides an important level of adaptability. The architecture also provides a flexible communication agent that can communicate with a user using a variety of communication media to meet the user's needs at any given point in time.**

Keywords-multi-agents; decision-support system;knowedge-based system; human-computer collaboration.

I. INTRODUCTION

While every control system is unique in terms of its needs, many control systems share a common characteristic; They need to gather input from a user, manage a situation based on the user's needs, monitor their environment to update the control plans as needed, and communicate with the user to inform them of any changes and update the plans based on the user's preferences.

While several general purpose agent architectures exist, in the work developed here, we custom-designed an agent architecture that balances our need for efficient execution and flexibility to customize

In Section II, we present the current state of multiagent architectures, in Section III, we present the basic architecture we developed for control systems, and in Section IV, we present two different applications we developed using our architecture. These applications include an agricultural management system with weather monitoring [1] and a call routing system for a teachnical support center [2]. A general conclusionis provided in Section V.

II. STATE OF MULTI-AGENT ARCHITECTURES

Multiagent architectures are architectures that utilize multiple cooperative autonomous agents that collaborate on decision making and coordinating their actions to accomplish tasks beyond their individual capabilities [3], [4]Communication styles among these agents can be hierarchical, centralized, or distributed[5],[6]. Over the years, specialized languages have been developed for agent programming [7]. The research reported in this article does not utilize any of these languages.

Most new developments in agent architectures are driven by application needs. Application domains are varied including areas like traffic management[8], [9], cooperating robots[10], supply chain management [11], and smart electric grid management[12]. The research reported here follows this trend by customizing the agent architecture to the needs of the application domains. We elected to do our own development rather than rely on a general purpose architecture for a more efficient execution and a higher level of flexibility.

III. BASIC ARCHITECTURE

Our control architecture is divided into four basic modules with an additional weather data gathering module that we found to be useful in several of our developed applications. These modules are the User Input module, the Knowledge-based System, the information delivery module, and the Timer module. These modules are connected centrally to a database. The database we used is Microsoft Access and the system has also been tested with SQL Server thus making it more flexible. The use of an intelligent-agents approach has allowed the development of a user-friendly system that can be applied to many monitoring systems.

A. System Features

- Allows the users to enter their information for performing daily activities
- Stores the information in a database
- Agent parses the information and stores it
- Can connect to the web and collects the weather information
- Expert System integrates with the agents and accesses the user database to make intelligent decisions
- Agents collect the information from the Expert System
- Agents convert the information to appropriate forms as needed for user communication, such as Text or Wave for voice communication
- Telephony Application Programming Interfaces (APIs) can call the users with the data stored in the Agents

 ISBN: 978-1-68558-046-9

using standard telephony or any of an array of Voice Over Internet Protocol (VOIP) applications

- Users are contacted using their preferred communication method (voice, text, email) and within each method, they are given the capability of interacting with the system.
- The system is maintained by a timer, which runs every morning by default and checks the data for the users. Users can customize the timer schedule.

B. User Input Module

The user interface is a critical part of any solution and should be very user friendly. We developed a web-based user-input module. The User can enter their weekly information in the system. The system is smart and updates its information every time a user comes into the system. The user is allowed to make their personal profile, which is password protected. Options provided for the user are customized based on the type of application such as farming, or home automations as discussed in the next section. The system also collects the user's preferred contact method and their needed credentials so they can be contacted as needed using the information delivery module. The data stored in the database is used by the expert system module to compare the data and generate appropriate results. These web pages are running on an Internet Information Server and are connected to the database.

C. Knowledge-Based System Module

Knowledge-basedsystems (sometimes referred to as Expert Systems) are computer systems, which provide expert quality advice, such as diagnoses, and recommendations given real-world problems. They are intelligent systems, which have knowledge stored in them and can make decisions, which normally require human expertise. It receives as input a problem and through its knowledge base makes decisions to give a solution to the problem. Knowledge-based Systems have been used in different areas such as medicine, robotics, mathematics, and various other fields. We include a generic template for a knowledge-based system in our architecture. This template needs to be customized for each application as discussed in the next section.

The knowledge-basedsystem is designed to access the central database and check the requirements for every user that were entered through the user interface and based on the given rules, it makes its decisions using the information stored in the Data Base Management System(DBMS) agent.

The Knowledge-based systemis triggered based on the schedule stored in the timer module to avoid unnecessarily using the system resources and skips processing the users who are inactive in order to minimize the resources used by the agents, the CPU time, and the memory. The knowledge-based system invokes the information delivery agent discussed below to alert the user as needed.

D. Information Delivery Module

The information delivery module is another crucial part of the system. This module is responsible for delivering the results; the useful information generated from the expert system to the user and acquiring any feedback from the user. There are diverse ways of communication available today. Multiple agents are provided and can be activated according to the user's availability and wishes. Each agent is responsible for one communication method, an agent for agent-initiated phone communications, an agent for agent-initiated e-mail communications, an agent for user-initiated web-based communications, as well as an agent for VOIP communication that can be customized with any of several available VOIP applications (Skype, Messenger, WhatsApp for now but others can be added). This multi-modal communication provides efficient, dependable, and accessible interaction with the users regardless of their physical location.

This module manages the call processing. This is a multi-function module. It performs the function of opening the line, making a live connection, and then passing the data with a two-way interaction between the user and the automated program. This module goes through the list of users generated by the knowledge-based system and plays a file for each user.

This module also includes another set of agents known as sound agents. For every user, a sound agent is generated. These agents perform the function of converting any necessary information to be communicated to the user into wave files. We have used the Speech Application programming Interfaces to generate the wave files. These files are then played to the user who gets called by the call processing system.

E. Timer Module

This module controls the invocation of the knowledge-basedsystem module and if needed the Web data gathering, which in-turn triggers the information delivery module. We defined a default schedule that runs once a day This schedule can be customized to meet any specific user needs.

Figure 1 shows the overall diagram of the general control architecture with its components and shows their interactions.

 ISBN: 978-1-68558-046-9

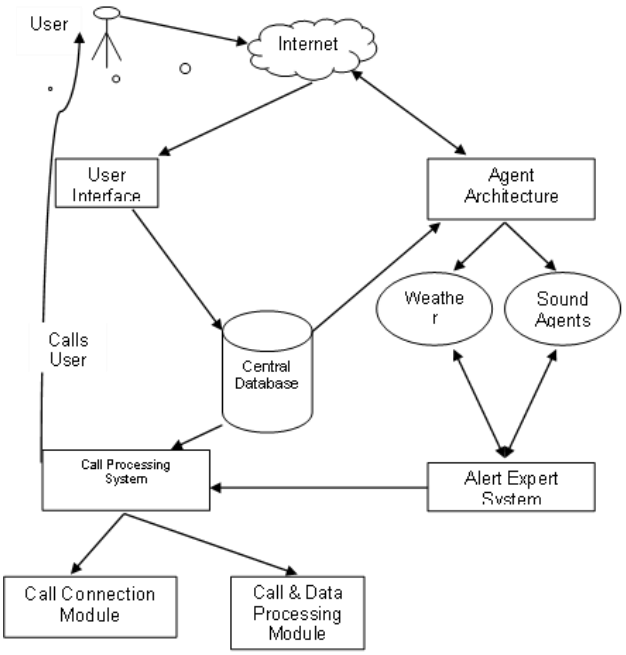

Figure 1. General Agent-Based Control Architecture

F. *Weather Data Gathering Module*

Since several of our applications rely on weather data. We designed a weather agent and included it with the basic architecture for use in any control situation that needs it.We included a collection of agents with one agent for each day of the week. While the choice of having an agent for each of the days of the week seems arbitrary, the choice was driven by our first application area in farming where we found that many of our famer collaborators are used to performing different tasks on different days of the week so having a different agent for each of the days of the week allows customization for each day in terms of the weather parameters needed. They use the web to collect information from weather web sites. They are also responsible for parsing the data and extracting the required data for their respective days. We developed these agents in Visual C++ and programmed them to use HTTP APIs to get the data. The Agent architecture looks for the web site URL, parses it and returns the type of service and its components. It then opens an HTTP connection and opens the source of the web page for data processing. The page contains the weather information for the given input parameters (location, time, needed weather data points).

There are seven weather agents, each holds a day's weather information and these intelligent agents in our system interact with the knowledge-based system. This interaction can occur in either direction. The knowledge-based system can trigger the weather agents if it needs weather information.The weather agents can also trigger the knowledge-based system if they detect a significant change in weather conditions.

G. *The Database*

There is another important agent, the DBMS Agent that retrieves the information from the database for the users. This agent gets updated during processing of different users. This agent works closely with the knowledge-based system and interacts with the above agents to facilitate decision-making. This agent carries the information for a specific user in a dedicated manner and holds that information until the decisions are created for that user.

IV. APPLICATIONS

We applied our control to several application domains. In this section, we present two of these applications; weather data monitoring for use in a farm control setting[13]and monitoring of incoming technical support phones for routing to appropriate technicians[14].

A. *Weather Monitoring*

Our weather monitoring system[14] is intended as a helper application to a farm management system we developed in the past to manage a wheat farm [1].

The agriculture community has always been dependent for their work on weather conditions, and it takes significant planning and money investment for them to perform their daily farming activities. If they are not well informed of the upcoming weather conditions, they are prone to revenue loss. Through this agent-based system, we provide this community an opportunity to prevent the waste of their resources and effectively utilize them by pre-informing them about valuable weather data based on their plans which we capture online in their personal accounts. There are seven weather agents, each holds a day's weather information and these intelligent agents in our system interact with the interactive decision criteria Alert Expert System, which activates these agents based on the management plans in the system. These plans include aspects such as irrigation and fertilization schedules which are typically extremely sensitive to changing weather conditions. Example conditions that the expert system module is trained to monitor include:

- Fertilization and Irrigation: Heavy Rains
- Spraying: Strong winds/Windy or rains
- Planting and Sowing: Rains

A top-level view of this system is shown in figure 2.

We tested this system with several collaborating farmers and all of them reported a high level of satisfaction with the alerts received from the system.

B. *Smart Call Routing*

In this application [2], we utilized our architecture to develop an agent system to route incoming technical support calls to appropriate technicians based on the needs of the users and the areas of expertise of the different technicians.

Traditionally, corporate computing systems consisted of hardware and software systems purchased from one or more vendors and maintained on site typically by local information technology staff. In recent years, a gradual shift occurred to a

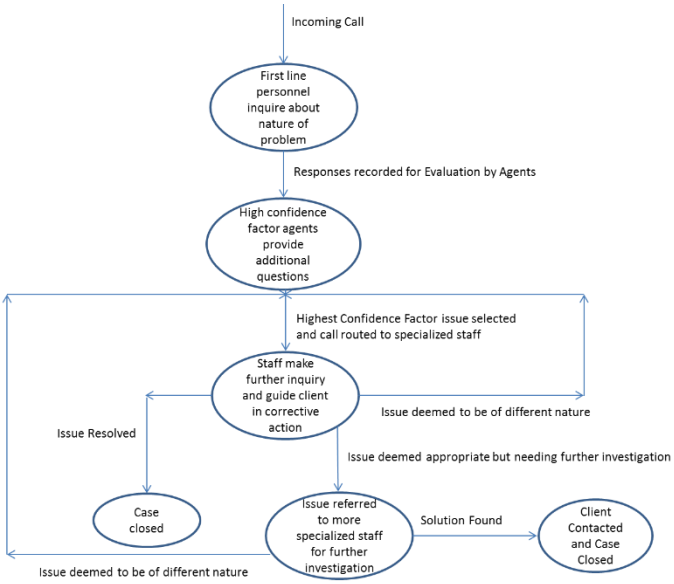

Figure 2. Top Level View of Weather Alert System

managed system model where corporations contract a vendor to install and support integrated IT systems. This shift was accompanied by a shift of the help desk support from the corporate IT department to the vendor's own staff. As a result, vendors set up large help desk installations where staff accept calls from personnel at a large number of corporate clients and attempt to troubleshoot a variety of issues.

The goal of this system is to assist the first line technicians in routing the calls to the appropriate service technicians. Figure 3 shows a top-level view of this system.

As reported in [2], this system resulted in the reduction of the average number rerouting incidents of incoming calls from 5.6 to 2.4 when used to replace an existing manual call routing system. This resulted in a higher level of satisfaction among surveyed callers.

V. CONCLUSIONS

In this paper, we presented a framework for communication agents embedded within an architecture for control based on a set of collaborating agents. The control architecture involves the users through communication between the users and the agent system. Multi agents are provided to enable intelligent decision making and interaction among users and their agents regardless of their physical location. We demonstrated the use of the developed architecture in two different control situations.The first application is for monitoring the web for the occurrence of an event such as a weather alert that would interact with predefined crop management plans. The other application is for routing incoming technical support calls to appropriate technicians based on their areas of expertise.

We are currently planning to apply this same architecture to other domain areas.

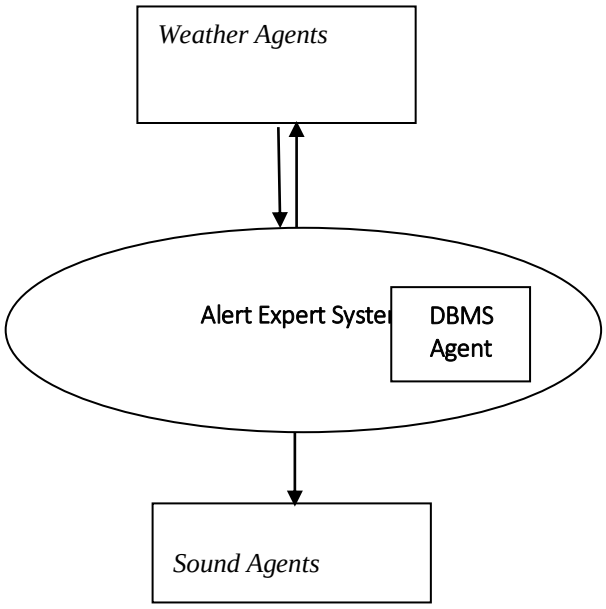

Figure 3: Top Level View of Call Routing System

REFERENCES

[1] A. Kamel, "Agent-Based Design of Agricultural knowledge Based Systems," International Journal of Computers and Their Applications, pp. 228-232, 4 2005.

[2] A. Kamel, "Smart Call Routing Utilizing a Multi-Agent Architecture," in The Eighteenth International Multi-Conference on Computing in the Global Information Technology, Barcelona, Spain, pp. 1-4 2023.

[3] P. Stone, and M. Veloso, "Multiagent systems: A survey from a machine learning perspective," Journal of Artificial Intelligence Research, vol. 11, pp. 216-252, 2000.

[4] M. Tambe, P. S. Rosenbloom, "Towards intelligent agent architectures: A survey.," in Firts International Conference on Multiagent Systems, San Fransisco, CA, 1997.

[5] W. Van der Hoek, J. J. Meyer, C. Witteveen, "Agent architectures and their evaluation: Who is afraid of a benchmark?" Artificial Intelligence, vol. 131, no. 1, pp. 181-213, 2001.

[6] J. F. Hubner, J. S, Sichman, O. Boissier, "Developing multi-agent systems with a FIPA-compliant agent framework.," Software: Practice and Experience, vol. 34, no. 2, pp. 159-189, 2004.

[7] M. Wooldridge, N.R. Jennings, "Intelligent agents: Theory and practice.," in First International Workshop on Agent Theories, Architectures, and Languages (ATAL'94), The Netherlands, 1995.

[8] J. Ma, L. Chen, Y. liu, "Multi-agent systems for traffic management: A review.," IEEE Transactions on Intelligent Transportation Systems, vol. 17, no. 2, pp. 308-320, 2016.

 ISBN: 978-1-68558-046-9

[9] J. Ma, L. Chen, Y. liu, "Multi-agent systems for traffic management: A review.," IEEE Transactions on Intelligent Transportation Systems, vol. 17, no. 2, pp. 308-320, 2016.
[10] M. Smadi, A. Kamel "A Knowledge Based Traffic Signal Control Application," in Fifteenth International Conference on Computer Applications in Industry and Engineering, San Diego, CA, 2002.
[11] A. Agah, G. A. Bekey, "Multi-robot assembly system using evolutionary multi-objective optimization and self-adaptive multi-agent systems.," Robotics and Autonomous Systems, vol. 43, no. 3-4, pp. 171-186, 2003.
[12] X. Li, D. Li, C. Huang, "A multi-agent architecture for intelligent supply chain management.," Journal of Systems Science and Systems Engineering, vol. 22, no. 4, pp. 445-466, 2013.
[13] T. Pinto, H. Morais, Z. Vale, P. Faria "Multi-agent systems for integration of distributed energy resources in power systems: Models, applications, and future trends.," Renewable and Sustainable Energy Reviews, vol. 49, pp. 345-357, 2015.
[14] D. Seth and A. Kamel, "An Intelligent-Agent Based Automated Weather Alert System," in Third International Conference on Computer Science, Software Engineering, Information Technology, E-Business, and Applications, Cairo, Egypt, pp. 48-63, 2004.

 ISBN: 978-1-68558-046-9