Association for Information Systems
AIS Electronic Library (AISeL)

AMCIS 1998 Proceedings

Americas Conference on Information Systems (AMCIS)

December 1998

The Information Warefare Advisor: An Architecture for Interacting with Intelligent Agents Across the Web

Carl Moore
Science Applications International Corporation

Jay Liebowitz
George Washington University

Robert Kilmer
LTC, U.S. Army War College

Robert Minehart
U.S. Army War College

The Information Warfare Advisor: An Architecture for Interacting with Intelligent Agents Across the Web

Carl R. Moore
Science Applications International Corporation
Software and Systems Integration Group

Jay Liebowitz
School of Business and Public Management
George Washington University

LTC Robert Kilmer and Robert Minehart
Center for Strategic Leadership
Science and Technology Division
U.S. Army War College

A Web Based Intelligent Advisor

The global spread of digital communications and automated information systems, and the ever widening dependence on these systems for critical economic, political and military needs, has made the protection of these systems, as well as the ability to attack an enemy's systems, an important part of military strategy and tactics. Information warfare (IW) has grown into a military specialty which seeks to use information and information systems to gain military advantages over an opponent. The US Army War College has several IW experts on its faculty, but during its annual Strategic Crisis Exercise (SCE) these experts are generally not available to assist students with questions on IW background or response planning since they have other duties during the exercise. In order to make the expertise of its faculty available to all students during each SCE, as well as preserve and expand this expertise as faculty come and go, a research project was established between the War College's Knowledge Engineering Branch and George Washington University to design and demonstrate an architecture for an on-line expert advisor in Information Warfare. The objective of the IW Advisor was to provide any SCE student or War College faculty member the ability to access the captured knowledge and expertise of IW experts through the college's private intranet using standard off-the-shelf web browser technology.

The IW Advisor Architecture – An Agent Brokerage

Figure 1 shows the logical architecture of the IW Advisor prototype. The distributed nature of the knowledge bases in the IWA environment led to the decision to base the IWA conceptual architecture on a distributed AI approach using a collection of cooperating intelligent agents. The IWA architecture used a network of agents, called an agency, with broker agents to handle problem decomposition and specialist agents to resolve the sub-problem portions assigned to them.

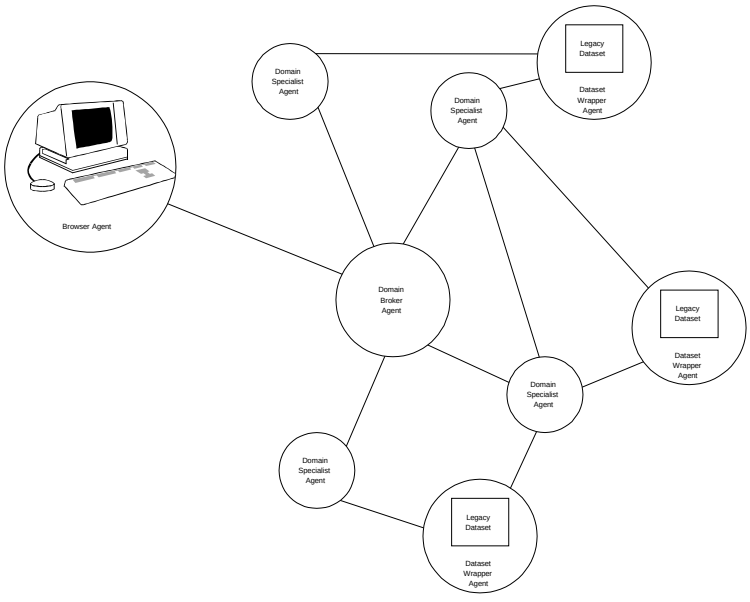

Figure 1. IW Advisor Brokered Agency Logical Architecture

The concept of the agency is to have a network of agents, each with a specialty, which will work together cooperatively to solve complex problems. The general problem solving paradigm is to decompose complex problems into portions each of the specialist agents can resolve, and then synthesize a complete response from the agent's partial responses. The domain specialist agents are the workers of the system. They contain the primary problem solution expertise. However their domain is very narrow. They solve a particular type of problem or portion of a problem. However rather than have the specialist agents acting autonomously to contract with other agents for sub-problems, broker agents were added specifically to handle problem decomposition

and agent sub-contracting. The broker agents act as generalists in their problem domain. Each broker in the system has knowledge of problem structure and solution for a given problem domain. The specialists accept subcontracts from the broker agents for a problem within their narrow specialty. Additionally wrapper agents were developed to provide interfaces to existing legacy data and text bases within the USAWC environment for use by the IW Advisor agents. This approach eliminated the requirement for all agents to be able to decompose complex problems, allowing specialist agents to be developed with the concentration on detailed problem solution, while the higher levels of problem solution in the form of problem decomposition and response synthesis was centralized within a few broker agents.

Results of the IW Advisor Development Project

The IW Advisor prototype was developed during the spring and summer of 1997, and demonstrated at the US Army War College in September of the same year. This prototype was used as a technical and conceptual demonstrator of the brokered agency architecture. The prototype system was implemented with a single broker and two specialist agents which together provide SCE students with advice and guidance in the processes of responding to information warfare events, or in developing a response plan to a specific IW event. This prototype was installed and demonstrated on the USAWC private intranet on an web server under the NT 4.0 operating system, with clients accessing the system via PCs attached to the network running either Netscape or Internet Explorer.

The design and development of the prototype Information Warfare Advisor presented technical challenges in three key areas, on top of the usual challenges of developing interactive knowledge based software. These three areas were the development of a scheme to provide the system with persistent memory during a dialog between the advisor and a user, the development of an Internet enabled inference engine for the advisor agents to communicate with users and with other agents, and the dual C++/CLIPS environment under which the IW Advisor was developed.

Conclusions

The IW Advisor brokered agency architecture was found to have both advantages and disadvantages.

Maintenance and expansion are more difficult than a network of hierarchically equal and fully autonomous cooperating agents since new agents must be somehow registered with one or more broker agents so the brokers know they exist and what their capabilities are. In like manner new broker agents must also be made known to other broker agents. This registration can be made manual at the time of adding or updating agents, increasing the load on maintenance programmers, or can be made automatic through logic included in the design of the agent architecture. However this increases the load on the agent developers, and indirectly on the maintenance programmers since now the agent designs would be more complex and hence more costly to maintain. In either event software engineering expertise is required to maintain or enhance the IW Advisor. It is not designed for updates by the user community.

Another potential disadvantage to this architecture is operationally it could become very resource intensive in terms of the computing power needed to operate. Since each agent instantiated during the course of solving a complex problem is essentially a complete inference engine running in its own memory space, a large agent system will require a large amount of memory if run on a single server. If the agents are distributed across a network then the resource requirements are also distributed, lessening the load on any one server, but increasing the communication bandwidth required on the network since the agents will be trading information over the network. The tradeoffs for implementing this architecture , as to centralized installation versus distributed installation, will vary from organization to organization.

There are however a number of key advantages offered by the brokered agency architecture that in most instances outweigh the disadvantages. The IW architecture's use of Internet enabled inference engines for the agents provides each agent developed for the system with all the flexibility and problem solving power of the CLIPS system. This can make for very powerful networks of cooperating agents. Also, as discussed previously, the use of the CLIPS system allows employing multiple agents but requiring only a single set of source code to maintain for all agents inference engines since the same executable is instantiated for each agent but with a different rule base for each agent. The use of CLIPS as a common inference engine for all agents in the IW Advisor system also makes the system expandable and extensible in capability. Additional agents can added by developing rule bases in CLIPS which define the agents' capabilities, and then identifying these agents to the appropriate broker agents for inclusion in the problem solving process. Also the CLIPS language is extensible by adding custom functions to the inference engine. And since only a single executable is used by all agents, modifying that one executable essentially automatically updates all agents with the language extension.

With a brokerage, cooperative agreements, or sub-contracts, among the agents are easier to establish. The brokerage concept reduces sharply the number of other agents a broker agent must be aware of in order to process a problem. Each broker acts as a generalist in its domain, providing a source of information on the specific agents which can be of assistance, or actually taking on the sub-task itself for solution. Brokers can also call upon other brokers, not just the specialist agents within its domain, thereby widening its field of expertise, as well as providing the system with a hierarchy of domain knowledge. Some brokers would have more general knowledge than others, each level of broker decomposing the problem into sub-tasks for subcontracting to other more specific brokers or to specialist agents. This structure provides the system with a large degree of modularity, making expansion easier to plan, execute, test, and manage.

References

Finin, T., and Fritzson, R., *KQML – A Language and Protocol for Knowledge and Information* Exchange, Proceedings of the Thirteenth International Workshop on Distributed Artificial Intelligence, Lake Quinalt, WA, 1994, pp. 126-136.

Hayes-Roth, *An Architecture for Adaptive Intelligent Systems*, Artificial Intelligence, Volume 72, 1995, pp. 329-365.

Lewis, Ted, *Internet as Metaphor*, IEEE Internet Computing, March/April 1997, pp. 94-96.

Maes, P., *Agents that Reduce Work and Information Overload*, Communications of the ACM, Volume 37, No. 7, 1994, pp. 31-40.

Maes, P., *Intelligent Software*, Scientific American, Volume 273, No. 3, 1995, pp. 84-86.

Marcenac, Pierre, *The Multiagent Approach*, IEEE Potentials, February/March 1997, pp. 19-22.

Norrie, D. H., and Gaines, B. R., *The Learning Web: A System View and Agent Oriented Model*, International Journal of Educational Telecommunications, Volume 1, No. 1, 1995, pp. 23-41.

Rechtin, Eberhardt, *The Synthesis of Complex Systems*, IEEE Spectrum, July 1997, pp. 51-55.

Riecken, D., M: *An Architecture of Integrated Agents*, Communications of the ACM, Volume 37, No. 7, 1994, pp. 107-116.

Russell, S., and Norvig, P., *Artificial Intelligence: A Modern Approach*, Prentice Hall, Englewood Cliffs, NJ, 1995, page 31.

Selker, T., *Coach: A Teaching Agent that Learns*, Communications of the ACM, Volume 37, No. 7, 1994, pp. 92-99.

Woelk, D., Huhns, M., and Tomlinson, C., *Uncovering the Next Generation of Active Objects*, Object Magazine, July/August, 1995, pp. 33-40.