Intelligent Mobile Agents in Military Command and Control

Susan McGrath, PhD
Lockheed Martin
Advanced Technology Laboratories
1 Federal Street
Camden, New Jersey 08102
(856) 338-4407
smcgrath@atl.lmco.com

Daria Chacón
Lockheed Martin
Advanced Technology Laboratories
1 Federal Street
Camden, New Jersey 08102
(856) 338-4196
dchacon@atl.lmco.com

Kenneth Whitebread, PhD
Lockheed Martin
Advanced Technology Laboratories
1 Federal Street
Camden, New Jersey 08102
(856) 338-4060
kwhitebr@atl.lmco.com

ABSTRACT

Lockheed Martin Advanced Technology Laboratories has been designing and implementing intelligent mobile agent prototypes for various military applications since 1995. Through our experience working in the military domain, we have identified a number of agent capabilities that are common themes in many of our applications, including information push, information pull, and sentinel information monitoring. We have implemented reusable agent components to enable rapid development of agent-based applications, where information push, information pull, and sentinel information monitoring are desired behaviors. We have gained a number of valuable insights about mobile agent development, deployment, and agent capability requirements in military applications. These lessons learned can be applied to other domains that have characteristics similar to the military applications on which we have worked. These characteristics include constraints on network reliability and bandwidth, domain-dependent information processing, and complex, autonomous information processing involving large heterogeneous data resources.

Keywords

Mobile agents, agent architectures, information agents, information discovery.

1. INTRODUCTION

Lockheed Martin Advanced Technology Laboratories has been involved in over a dozen intelligent mobile agent programs supporting all branches of the military[1], beginning with our work on the Domain Adaptive Information System (DAIS) [6], [13], which was supported by the Defense Advanced Research Projects Agency (DARPA).

LEAVE THIS TEXT BOX IN PLACE AND BLANK

Based on our initial experience with DAIS and subsequent work developing agent-based systems for military applications, we noted that many of these applications had similar operational models and goals and, therefore, required development of similar agent capabilities. The Advanced Technology Laboratories developed generalized notions of three of these capabilities: information push, where agents automatically send information to other agents or entities that may need it; information pull, where agents retrieve relevant information from distributed sources; and sentinel monitoring, where one or more agents persistently checks for an event or existence of a condition and reacts to its occurrence.

We have recently completed an effort to synthesize our agent components into a single development framework that will enable us to rapidly construct agent-based applications capable of exhibiting a wide range of agent behaviors, including information push, information pull, and sentinel-style monitoring. We found these three behaviors to be needed, either individually or in combination, in nearly all of the military applications we have addressed in the past five years.

In this paper, we present three examples of military command and control (C2) applications to illustrate what is meant by agent-based information push, information pull, and sentinel behavior. We then present agent system components designed and implemented as part of our capabilities synthesis effort. These capabilities support rapid development of agent-based applications, including those where the aforementioned agent behaviors are desired. To illustrate the use of these components and our implementation approach, we provide specific details of a C2 mobile agent application that demonstrates information retrieval and sentinel monitoring. Finally, we focus on valuable lessons learned from developing agent systems for military use and from our interaction with the larger agent community. These insights pertain to agent system development, agent system deployment, and agent capability requirements in military applications. We discuss them in the hope that they can be of use to others developing agent-based systems with similar requirements, e.g., constraints on network reliability and bandwidth, domain-dependent information processing, and complex, autonomous information processing involving large heterogeneous data resources.

2 AGENT UTILITY IN COMMAND AND CONTROL

Information Pull. Our first endeavor in intelligent mobile agents, DAIS, was a distributed information retrieval system that used mobile agents to query heterogeneous databases over intermittently connected and low-bandwidth networks. DAIS was deployed with the U.S. Army 201st Military Intelligence (MI) brigade at Fort Lewis for over a year. It significantly reduced information dissemination and retrieval latencies, enabling MI operators to radically improve the reliability of their analyses. DAIS provided more than just simple queries; it allowed a user to make "abstract queries," information requests for which the user could specify query parameters in high level concepts rather than in exact database schemata. The abstract query mechanism provided mobile agents with tasks to be executed at individual databases. With this capability, operators could avoid exhaustive searches of all data sources and instead search the 2% (average) of sources that were actually relevant to the query. DAIS is a deployed technology proven to contribute significantly to unit effectiveness.

Information Push. On DARPA's Small Unit Operations (SUO) program, we used intelligent agents to assess threat to and disseminate reports among deployed small units. Soldiers in the field may receive reports from "organic sensors" and may generate reports based on their own observations. One goal of the SUO program was to rapidly push critical information gathered by deployed soldiers to other squad members and to echelons above so that it could be acted on as quickly as possible. We used agents to disseminate high priority reports to upper echelons and to other soldiers in a unit based on their information needs. To minimize load over the very low bandwidth network, an analysis agent determined which soldiers in the squad needed a given piece of information based on knowledge of each soldier's location and determining if the information was relevant to that location. The delivery agents distributed the reports and handled connection failures by retrying at intervals, informing the sender of a severed connection upon unrecoverable failure. Mobile agents fulfilled an important requirement on the SUO program: robust information dissemination across unreliable networks (Figure 1).

Sentinel Monitoring. For the U.S. Air Force Air Mobility Command (AMC), we used agents to discover and alert on conflicts in air mobility plans [2]. The AMC's mission is to provide airlift, air refueling, special air mission support, and aeromedical evacuation for forces. AMC manages several types of missions. Channel missions are standard supply routes that recur on a consistent basis; they are typically planned one to three months in advance. Contingency missions are critical emergency operations that take precedence over channel operations and that may be planned only a few hours in advance. High priority contingency missions often supersede plans for channel missions, and air planners do not currently have the ability to monitor for mission conflicts after missions have been submitted to the air plan database. The prototype that we developed for the AMC used sentinel agents to persistently monitor distributed data sources for potential conflicts to a mission plan, to perform analysis to determine if a conflict actually occurred, and then to alert air planners so they could replan the mission to avoid the conflict. This sentinel prototype enabled users to passively monitor critical data sources and automatically receive relevant alerts, thus eliminating the need for the user to be constantly involved in time consuming and laborious data searching tasks.

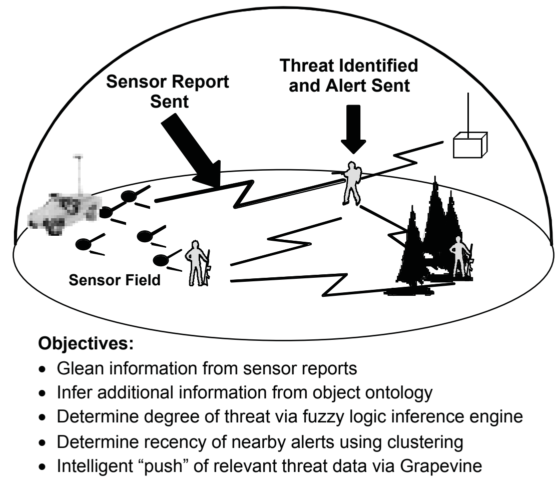

Figure 1. Static and mobile agents developed for Small Unit Operations assessed threat and disseminated information to individual soldiers.

3. AGENT SYSTEM DEVELOPMENT COMPONENTS

Our work on DAIS, SUO, and AMC, among other programs, convinced us of a widespread need for efficient, robust information retrieval, dissemination, and sentinel monitoring. We have found that mobile agents provide an excellent means of meeting requirements for many military applications, particularly those in which low bandwidth and/or intermittently connected networks are present. Agents are also useful tools when information to be discovered and/or processed resides in heterogeneous, distributed databases. A discussion of requirements that agents may meet across many domains is provided in [9]. We have developed a number of components that enable us to build agent-based applications using a standard development framework, which we describe in the following section.

3.1 Extendable Mobile Agent Architecture

The majority of our work in intelligent agents is built on a Java-based Extendable Mobile Agent Architecture (EMAA) [10]. EMAA is an architecture specification; its implementation is object-oriented and highly extendable. It provides a simple way for a mobile agent to migrate from one computing node to another and to use the resources at that node.

Figure 2 depicts the basic architectural components of EMAA. At EMAA's core lies an agent Dock that provides an execution environment for agents, handle incoming and outgoing agent transfers, and allows agents to obtain references to services. EMAA allows users to define agents, services, and events. Agents are composed of small, reusable tasks performed to meet a goal for a user. An agent's tasks are encapsulated in an itinerary. We have used a number of itinerary structures, including sequential lists and more flexible finite-state machines. Agents may be mobile and they typically make use of stationary services. Services may implement connections to external systems (e.g., databases, other applications), may themselves provide some complex functionality, or may carry out any number of other functions, so long as they

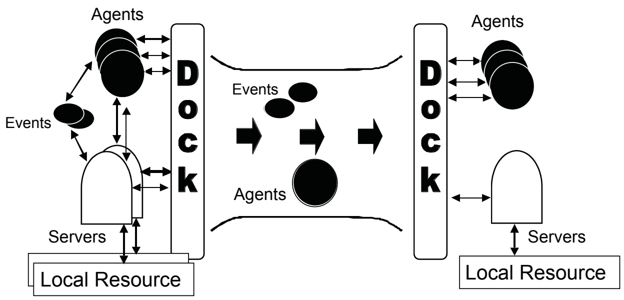

Figure 2. The basic EMAA components: the Dock, agents, servers, and events.

are not themselves primary actors. Goal-orientation and directed activity is generally reserved to be the function of agents. Both agents and services may send and receive events.

Agents, servers, and Docks form the basic EMAA components, but there are other optional, predefined servers that can prove extremely useful when creating distributed solutions, most notably the Class Loader and Resource Servers.

Class Loader. When an agent migrates to a node, there may be some components of the agent already present at the destination node. It is more efficient for an agent to migrate with only the references to its components and use those present at the destination. When the dock is ready to allow an agent to execute, it ensures that all software components required by the agent are present. If any component is not found at the destination node, then the dock can request the *class loader* to obtain that software component from the last node the agent came from that is known to have the missing component. This server allows EMAA to minimize the size of agents sent over communication links, thereby reducing bandwidth load.

Resource Servers. A resource server advertises a node's resources to all other nodes. This allows every agent to be aware of the tools available on the network. Using a resource server, an agent can determine what software or data resources it can use to achieve a goal at runtime, thereby facilitating inter-node resource sharing.

3.2 Pattern Development Layer

We created a pattern development layer [2] that allows programmers to reuse code to produce abstract behaviors, such as information push, information pull, and sentinel monitoring. The pattern development layer assumes the use of finite-state-based itineraries because complex behavior that engenders a need for patterns cannot be easily represented with simple sequential itineraries. Using agent itinerary behavior patterns with a library of commonly used tasks drastically reduced our agent application development time to 40% in some cases (indicated by multiple project lead estimates). This approach reduced redundant creation of agent behavior and it increased the simplicity of agent itinerary creation while preserving agent flexibility and decision-making capability. To date, we have created two sentinel-monitoring patterns that have been widely used on our agent programs. Detailed discussion of these patterns can be found in [2]. We foresee additional patterns emerging in information retrieval, dissemination, and monitoring as we continue to design and develop agent applications.

3.3 Distributed Event Messaging System

We found in our development efforts that it is beneficial to monitor and control agent activity in an application. For instance, if an agent has been tasked to persistently monitor a data source, it may be desirable to change its persistence parameters or halt the agent activity if the monitoring is no longer necessary. Communication among mobile agents raises problems that are not handled by standard distributed communication frameworks. For example, directed messages may miss a recipient because the recipient has moved. To address issues related to communication among mobile agents, we developed the Distributed Event Messaging System which uses event transceiver servers distributed across a network to permit control and monitoring of mobile agents [11]. These transceiver servers allow agents to register remotely for events by event source and event type. Events may be broadcast or unicast, and reliable delivery of unicast events is guaranteed through the use of event queuing on the receiving side. Use of this reliable communication mechanism allows one to remotely control agents; one can pause, suspend, resume, stop, or even retask mobile agents. We have also defined agent-monitoring events so that users may monitor the progress of an agent across a network and even through its itinerary.

4. AGENTS FOR THEATER AIR MISSILE DEFENSE

In this section we describe the Cooperating Agents for Specific Tasks (CAST) prototype system developed for the U.S. Navy with support from DARPA. CAST applies the tools described in Section 3 to distributed, information retrieval and monitoring problems that surface frequently in Theater Air Missile Defense (TAMD) operations.

TAMD operations rely heavily on Command, Control, Communications, Computers, and Intelligence (C4I) to support active and passive defense as well as attack operations. C4I's support role is a critical one because TAMD operations must often be planned and executed in a matter of minutes; this urgency leaves little time for slow, unwieldy queries, as well as the data analysis that must take place to sublimate huge volumes of data into meaningful information. Today, even at the best-equipped command center, the TAMD process suffers gravely from widespread use of stovepipe systems, loosely coupled command centers, understaffing, and too much information retrieval and analysis to perform in too little time. In short, far too often people are in the process between complex systems and, when the people became too overloaded, valuable information is unavailable when it would be most useful.

CAST addresses the problem of discovering and prosecuting mobile, time-critical targets, particularly missile transporter/erector/launchers (TELs). TEL units launch ballistic missiles from ground vehicles; they operate out of "hide sites," camouflaged areas from which they can quickly be moved to a firing position. In combat situations, information that could be combined to determine the likely locations of hide sites and prevent missile launches often exists but it is spread across several remote military and open-source databases. Currently, there is rarely time to obtain and systematically analyze these data at the desired level of detail; if this capability existed, it could significantly buttress the military's ability to deal with theater ballistic missiles. In CAST, intelligent agents persistently monitor multiple, remote, heterogeneous data sources, perform analyses, and alert an intelligence operator to possible hide sites.

The technical problems involved in finding indications of TEL activity are significant: multiple indicators from distributed, heterogeneous sources must be correlated and combined over time; data must be filtered at the source to reduce load on the already overtaxed SIPRNET (military internet) or low-bandwidth wireless networks; and the search for TEL activity must occur persistently because data at most sources is frequently changing. Manning, space, and time constraints currently make it difficult for human operators to perform this process; a human needs intimate knowledge of applicable databases and a significant chunk of time to manually perform the necessary search, analysis, and monitoring. Using CAST, an operator can select a geospatial region to monitor for TEL activity, start a search, and perform other duties until alerted that probable TEL activity has been detected.

CAST takes an approach similar to that of an intelligence analyst in searching for indicators of TEL movement or activity. An analyst would probably monitor data sources likely to give at least preliminary relevant information. If they spot some possible indicator, they may search other, potentially remote databases for reports that they can correlate to the indicator they found. Because some relevant databases are periodically updated, they might repeatedly check those data sources until they find enough evidence to confirm or deny the hypothesis. Unfortunately, because of the volume of data to be examined, a human analyst would have difficulty monitoring more than a very few indicators at once.

Figure 3 depicts the information flow and agents used in CAST. CAST uses mobile agents to filter through volumes of data, increasing the number of indicators that can be analyzed in realtime. Much like a human analyst, CAST views each distinct possible indicator as a hypothesis that should be confirmed or eliminated based on further analysis. Each hypothesis is independently followed; data related to the hypothesis is collected and is evaluated to assess validity. CAST assigns an agent, or "Scout Agent," to each hypothesis. The Scout Agent carries its hypothesis around the network, searching distributed Whiteboards (information sharing structures) for correlating data, and evaluating the likelihood that the correlated data indicates a TEL hide site. In the meantime, Information Agents, agents specialized for information retrieval, search relevant databases for information that could possibly be related to TEL activity and post results to shared Whiteboards to be examined by the Scout Agents. To determine if the criteria fulfilled by a set of reports are sufficient to generate an alert, the Scout Agent runs the criteria against a fuzzy rule evaluation engine. If there is sufficient information to determine that the condition is true (high likelihood of a TEL hide site at a location), then an Alert Agent is created to inform the querying node.

Because data is filtered where it is found, comparatively small amounts of data are sent over the military computer network. The agents used are very lightweight, taking data to processing resources that would be too large to reasonably transport over the network. The use of Whiteboards to facilitate agent information sharing conserves processing resources by eliminating the need for multiple agents to make the same database queries. In short, CAST is a system that efficiently and persistently monitors multiple, distributed data sources, analyzes discovered information, and alerts an analyst when a critical condition exists—tasks that cannot currently be accomplished by a human analyst in realtime.

We have demonstrated the CAST prototype in three U.S. Navy Fleet Battle Experiments (FBE). For the second FBE, we installed CAST aboard the U.S.S. Coronado, the Third Fleet command ship, to support TAMD operations in the Joint Air Operations Center. There was insufficient time to integrate CAST with live data feeds, but CAST did receive realistic data that corresponded to actual experiment events and that was synchronized with the

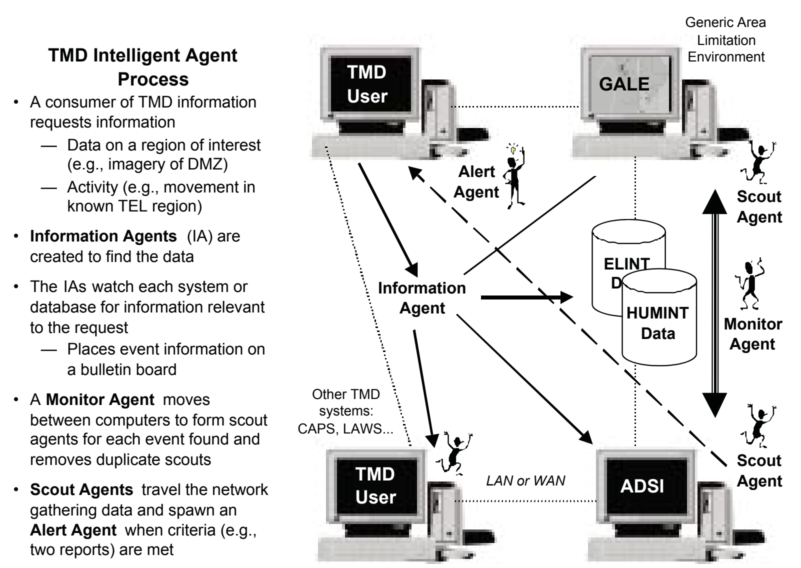

Figure 3. CAST mobile agents persistently monitor for new data and correlate it to find time critical mobile targets.

progress of the experiment. CAST retrieved this data over SIPRNET from onship and offship data sources to support abstract queries and TEL activity searches. The Navy actually used CAST's alerts on potential TEL locations in the tasking of sensor and strike assets. This represented a resounding success for the Advanced Technology Laboratories because it demonstrated that autonomous, mobile agents could be exploited to perform time-critical retrieval, monitoring, and analysis functions. The enthusiasm of personnel in the Navy's air operations center for CAST during the FBE highlights the need for mobile agents in the C2 environment.

5. LESSONS LEARNED

The majority of this paper concentrates on illustrating the need for information retrieval, dissemination, and monitoring in military applications, and on describing our efforts to address that need. However, we also wish to share other lessons learned from our experiences designing, implementing, and operating agent-based systems for the military. In this section, we address three areas that we feel may provide insight to those considering the applicability of agents to military domains. First, we discuss requirements for deployable systems that have emerged from the suggestions of users and from our own experiences. Second, we present lessons learned in developing and deploying agent systems to military environments. We close with points regarding the current state of intelligent agent technology.

5.1 Critical Military Requirements

Information Retrieval, Dissemination, Monitoring. A clear need has been demonstrated for information retrieval, dissemination, and monitoring across multiple military applications. This functionality has been effectively achieved given the current state of mobile agent technology. Such agent-based systems will be most effective if they can be implemented as extensions to systems that are already in use, eliminating the requirement for already overloaded operators to learn more new information systems. We have also seen the need for information retrieval, dissemination, and monitoring in business software systems, particularly those incorporating legacy systems, demonstrating the widespread need for such capabilities.

Heightened Efficiency at Low Bandwidths. Mobile agents can make an enormous impact on low-bandwidth military networks because they can analyze and filter data at its source, thus reducing the amount of bandwidth required to execute a task. This is particularly pertinent to military networks in the "last mile"; networks at brigade and battalion echelons tend to be fairly high-speed, but communications links to deployed teams tend to have much lower available bandwidth.

Robust Performance on Intermittently Connected Networks. Wireless networks often form the basis for deployed squad communication; such network connections frequently fail as squad members move out of range. Because mobile agents are able to persist at a node, even without a fully connected network, they provide a robust level of fault tolerance in unreliable networks. In addition, it would be beneficial if software deployed at the lowest echelons could flexibly react to such network failures and reconfigurations, dynamically rerouting to find a connection to the desired destination. It seems natural to think of different mobile agents implementing different recovery and rerouting policies based on their goals and needs. Work in this area is being conducted by Dartmouth College on a Multi-University Research Initiative on Transportable Agents for Wireless Networks [1].

5.2 Mobile Agent System Development

Agent Behavior Patterns. We have successfully demonstrated the utility of using agent behavior patterns to enable rapid application development with reusable code. The abstraction of additional behaviors and implementation of corresponding agent patterns would further our ability to develop agent-based systems capable of exhibiting complex behaviors. We think agents capable of learning are of particular interest.

Opportunistic Cooperation. Collaboration among mobile agents can be beneficial in applications, e.g., to introduce parallel execution of tasks or reduce redundant operations. However, true collaboration is difficult to achieve due to the lack of a standard means of representing tasks and data. Ontologies and semantic definition languages provide the capability to do this, but there has not been widespread agreement on standards for these in the military. In lieu of full collaboration, we have often used what we call "opportunistic cooperation"—in the form of Whiteboards—to allow agents to share data [8]. We find that in many situations, this reduces the number of redundant queries and amount of bandwidth used.

5.3 Mobile Agent System Deployment

Rapid Prototyping. Cyclical rapid prototyping is an effective development strategy for transitioning agent systems to users. On the DAIS program, we were able to develop a close and continuing relationship with users; we participated in at least 12 exercises and made countless changes to reflect user requirements. This resulted in a product that became a part of the brigade's tactics and procedures [12]. Of course the success of a rapid prototyping approach relies on the consistency and availability of domain experts and end users. On the CAST program, we took a different approach because a different fleet hosted each FBE; hence, a different set of users was involved. Although this scenario prevented us from executing the cyclic rapid prototyping approach in both FBEs, we were able to produce useful agent applications that generated much enthusiasm among operational experts.

Explanation Facility. Our development efforts show that a well-known expert systems "nugget" holds true; if users are to trust a system, the system must absolutely have an explanation facility. It is tempting to create highly complex, "intelligent" agents that reason for the user; this may be interesting science, but it may not in itself meet the needs of a military user. It is critical for an agent, when providing analyzed data, to accompany it with the raw data that was critical in making the decision.

5.4 State of the Technology

We perceive that the trend in military agent applications will move from limited applications performing isolated functions to collaborative applications interacting with other systems and data sources to reach complex goals. However, agent technology today is still struggling to gain acceptance in military domains. The Integrated Marine Multi-Agent Command and Control System

(IMMACCS) [7] is a notable exception. IMMACCS is an ambitious, squad-level, C2 decision support system developed by CAD Research Center and supported by the Marine Corps Warfighting Laboratory. It is designed with a fairly comprehensive object model and an open architecture that would allow agents from other agent systems to subscribe and make use of objects and events. However, on the whole, industry must first prove to the military the utility and reliability of agents. This is best accomplished by fielding successful focused applications. As multiple agent applications emerge to provide significant functionality, there will be impetus for new agent and non-agent systems to interface and collaborate with them, thus prompting an increase in acceptance of such systems by military users.

Several technical requirements must be met to support widespread transition of agent technology to the military domain. Robust agent behavior control mechanisms must be developed and emplaced; the military must be convinced that agents are controllable tools rather than dangerous (in terms of security and bandwidth) and uncontrollable viruses. Conversely, resource control mechanisms must be instituted because agents have greater potential than human operators to overload legacy resources. For their part, military network designers must carefully define data and legacy-system access and release policies, so that agents are not in danger of inadvertently disseminating sensitive information. From an agent technology standpoint, perhaps the most outstanding need is for a standard agent capability and data description language—a semantic framework that supports collaboration across many heterogeneous agent systems. DARPA's Control of Agent Based Systems program has put in place a JINI-based software infrastructure to support agent control and collaboration [3], [4]. The DARPA Agent Markup Language program is creating an XML-based language as a potential solution to the semantic interoperability problem [5]. These efforts are moving the state of agent technology further along the trend toward collaborative agent systems for military applications.

6. ACKNOWLEDGMENTS

We would like to thank Julius Etzl and Martin Hofmann for their technical and editorial contributions to this paper. Portions of the work described herein have been supported under the following contracts: DARPA contract N666001-95-C-8636, DARPA contract F30602-98-C-0162, DARPA/ITT contract PO # 9920225, and DARPA/ITT contract PO # 1800348.

7. REFERENCES

[1] Caripe, W., Cybenko, G., Moizumi K., and Gray, R. "Network Awareness and Mobile Agent Systems," *IEEE Communications Magazine*, 36(7): 44-49 (July, 1998).

[2] Chacón, D., McCormick, J., McGrath, S., and Stoneking, C., "Rapid Agent System Development Using Agent Itinerary Patterns." Submitted to the Second International Conference on Agent Systems and Applications and Fourth International Conference on Mobile Agents (ASA/MA 2000), Zurich, Switzerland (September, 2000).

[3] Control of Agent Based Systems (CoABS) website. http://dtsn.darpa.mil/iso/programtemp.asp?mode=126

[4] CoABS Technical Coordination website. http://coabs.globalinfotek.com/

[5] DARPA Agent Markup Language website. http://dtsn.darpa.mil/iso/programtemp.asp?mode=347

[6] Hofmann, M.O., McGovern, A., Whitebread, K.R., "Mobile Agents on the Digital Battlefield," *Proceedings Second International Conference on Autonomous Agents (Agents '98),*, Minneapolis/St. Paul, 219-225 (May, 1998).

[7] IMMACCS (Integrated Marine Multi-Agent Command and Control System) web site. http://www.cadrc.calpoly.edu/Projects/immaccs.html

[8] Kay, J., Etzl, J., Rao, G., and Thies, J., "The ATL Postmaster: A System for Agent Collaboration and Information Dissemination," *Proceedings Second International Conference on Autonomous Agents (Agents '98)*, Minneapolis/St. Paul, 338-342 (May, 1998).

[9] Lange, D., and Oshima, Mitsuru, "Seven Good Reasons for Mobile Agents," *Communications of the ACM*,Vol. 42(3), 88-89 (March, 1999).

[10] Lentini., R., Rao, G., Thies, J., and Kay, J. "EMAA: An Extendable Mobile Agent Architecture," Fifteenth National Conference on Artificial Intelligence Workshop on Software Tools for Developing Agents, TR WS-98-10, Madison, WI, 133-34, (July, 1998).

[11] McCormick, J., Chacón, D., Lentini, R., McGrath, S., and Stoneking, C., "A Distributed Event Messaging System for Mobile Agent Communication." Submitted to the Second International Conference on Agent Systems and Applications and Fourth International Conference on Mobile Agents (ASA/MA 2000), Zurich, Switzerland (September, 2000).

[12] Tactical HUMINT Battalion Tactics, Techniques, and Procedures. http://call.army.mil/call/trngqtr/tq198/humint.htm

[13] Whitebread, K.R. and Jameson, S.M., "Information Discovery in High-Volume, Frequently Changing Data," *IEEE Expert*, 51-33 (October, 1995).

8. FOOTNOTE

Programs for which the Advanced Technology Laboratories has developed intelligent agent technologies include:

- DARPA's Control of Agent-Based Systems (Multi-Agent Common Operating Environment)
- DARPA's Control of Agent-Based Systems (Cooperating Agents for Specific Tasks)
- DARPA's Human Computer Interactions (Domain Adaptive Information System)
- DARPA's Small Unit Operations (CyberAngels)
- DARPA's Joint Logistics Advanced Concept Technology Demonstration (CyberXpress)
- DARPA funding for the Domain Query Ontology
- U.S. Communication-Electronics Command's Logistics Command and Control Advanced Technology Demonstration
- DARPA's Listen Compute Show-Marine
- DARPA Communicator
- U. S. Air Force Rome Laboratory's Agent–based Decision Aids for Mobility Operation