

# Facilitating Open Communication in Agent Systems: The InfoSleuth Infrastructure

Marian H. Nodine\* and Amy Unruh\*,†

\* MCC, 3500 West Balcones Center Drive, Austin, TX 78759  
{nodine, unruh}@mcc.com

† Systems and Software Lab, DSPS RDC, Texas Instruments  
PO Box 655303, MS 8378, Dallas, TX, 75265

**Abstract.** This paper addresses issues in developing open multiagent systems, in which it is easy to expand the functionality by adding new agents with new capabilities, and which facilitate interoperability with other agent systems. We argue that an open multiagent system should define the following support elements for agent communication:

1. A common set of *speech acts* to define the types of messages that an agent might send to another agent.
2. A common *service ontology* by which the agents can describe their capabilities to each other, and reason about which agents have the capabilities needed to execute specific tasks.
3. A common set of *prescriptive conversation policies* to define the acceptable exchanges of messages between agents.

In addition to the above, we also discuss the utility of having a matchmaking agent that can reason over agent capabilities to recommend agents for specific tasks, where the capabilities and requirements are defined using a common service ontology. This ensures that the semantics of matching agent capabilities to task requirements remains the same across the multiagent system.

## 1 Introduction

Autonomous, intelligent agents are currently being developed in many different application areas. In a multiagent system, each agent executes specific types of tasks, and serves a specific purpose. No agent does an entire job. Rather, it does what it can, then delegates tasks to other agents. No one agent has control over how a task is executed or who executes it, other than controlling the execution of its own specific subtasks.

Because of their modular, cooperative and distributed nature, it should be easy to expand the functionality of a multiagent system by adding new agents that have different capabilities. Ideally, these agents should be able to be added by constructing them independently of the multiagent system, then providing some means by which the multiagent system can find out about the agent and its capabilities. Another attractive option with respect to agent systems is to allow different but related multiagent systems, developed independently, to interoperate.

An open agent architecture is designed to facilitate the addition of new agents to a multiagent system, and to facilitate the interoperation of related multiagent systems. In

an open agent system, the agents operate within a common infrastructure that governs how the agents advertise their capabilities and services, how they select which agent is appropriate for executing a given task at a given time, and how they communicate with one another during the execution of a task. This means providing all agents not only with a common understanding of specific messages and message types, but also a common understanding of the dialogs that can occur between pairs of agents, and among subgroups of agents that were specifically designed to work together. Furthermore, if agents are self-describing over some common service ontology that can be assimilated by other agents, this facilitates the integration of agents with new capabilities. In this paper, we specifically address issues involving inter-agent communication that impact the openness of a given multiagent system.

## 2 Keys to Open Communication

In a multiagent system, the need for agents to communicate is obvious. Several loose standards for agent communication have been proposed (e.g., KQML [7], FIPA [22], and even in some cases, CORBA). Furthermore, at least in the information integration area, several multiagent systems with sufficiently similar communication paradigms have evolved to the point where it is reasonable to start to attempt interoperation. Our agent system, InfoSleuth, is one of these. InfoSleuth supports a dynamic set of agents, spread out across a network, that collaborate to do information retrieval and integration.

Our experience so far has led us to appreciate the need for standard, open agent architectures to allow for the easy integration of new agents into an existing multiagent system, and to allow for the easy interoperation of related multiagent systems. Many multiagent systems are “closed”, either because they communicate using their own protocols, or they assume that each agent has specific, built-in knowledge of most other agents in their system. In this paper, we address several overall requirements that openness imposes on the architecture of a multiagent system. We will use our experiences with InfoSleuth to illustrate these issues.

We propose that keys to communication in open systems include a shared set of *speech acts* (“verbs”) to provide a structure to the discourse among agents, a common service ontology (a shared set of “nouns” and “adjectives”) which provides a dictionary of meta-information about agent capabilities, and a shared set of prescriptive conversational policies to provide a structure for basic agent dialog. The greater the extent to which this shared information is supported, the easier agent integration and interoperation will be. Also, these requirements are separate from the communication issues involved in the use of a shared domain ontology and domain communication language to execute domain-specific tasks.

### 2.1 Speech acts, or “verbs”

The concept of *speech acts*, or *illocutionary acts*, has grown out of philosophy and linguistics research [5]. These actions include requesting, promising, offering, acknowledging, asserting, etc. It is suggested that human utterances are the observable byproduct of such actions.