Chapter 1

WebBots, Trust, and Organizational Science

Kathleen Carley *Michael Prietula*

WebBots are artificial creatures. Now, by "artificial" we do not mean that they do not exist, for they do. In fact, we built some. Yet, WebBots are neither biological nor mechanical creatures; WebBots are computer programs. But computer programs of a very special type. WebBots are programs that help their human counterpart(s) to achieve goals and solve problems. What is unique about WebBots is that they do much of it on their own over webs of interconnected networks.

One of the major applications we see for WebBots is to be "intelligent explorers" on networks (including the Internet) for their human (i.e., corporate) counterparts. Thus an organization might have dozens, hundreds, or even thousands of corporate WebBots actively searching, communicating, traveling, and even reproducing over networks around the world for a wide variety of purposes. Simpler types of such creatures are being researched or even employed by firms such as AT&T, IBM, Apple Computer, Xerox, Microsoft, Hertz, Ford, and even the White House (Houlder 1994; Keller 1994). Although the specifics of any vibrant and emerging technology is extremely difficult to predict with certainty, the current trends in information technologies all point to a single, inescapable prediction: *the WebBots are coming*.

WebBots (or whatever you wish to call them) can take on a wide variety of forms. In this chapter, we will briefly mention some of these, but we are going to describe a different kind of WebBot. The WebBot we will describe will have very unique properties. To get these unique properties, we will propose a very unique architecture for WebBots. The interesting elements of the proposed architecture is that it provides a fundamental framework for general WebBot intelligence and permits a unique set of mechanisms for defining, measuring, and sharing corporate learning, memory and knowledge.

In this chapter, we first offer a brief look at WebBot-like programs. This is not a new concept; rather, we are building our approach on a long stream of incremental research from several different perspectives. We then present an architecture that can realize a specific type of intelligent WebBot agent. A WebBot agent that can reason and communicate with other WebBots. As a central point of this chapter is exploring the social aspects of WebBots, we next describe a computational experiment in which we simulate an organization of WebBots. In this experiment, we assign tasks to a small group of WebBots, adjusting and experimenting with a particularly important aspect of WebBot interaction – trust and forgiveness in information exchange. We conclude with a speculative discussion on the implications of applying an organizational science perspective to an organization of WebBots. Should we begin to define an organizational science of WebBots? Is it possible? We argue that this is not only possible (though certainly not easy), but essential, in order to successfully assimilate such technology (or technologies) into the corporate environment. We propose that the foundations for studying WebBot organizational science has already been formed.

About WebBots

In one sense, this chapter is quite speculative. The WebBot creatures of the type we are addressing are not quite ready for prime time – but close. They are in the digital Catskills of information technology: corporate and university laboratories. The WebBots in our world are related to digital creatures that go by many names, depending on their particular capabilities, or even on the particular laboratory or organization where they are being created. There is no commonly accepted definition for the *term* WebBot; however, the *concept* of a WebBot has emerged at various times over the past decades in both formal and informal settings.

We have witnessed lively discussions at our research conferences over who invented/used what term first, who actually constructed the first <fill in your term here>, and what stream of research was actually the most responsible for the current perspective(s). It is perhaps easiest to think of WebBots as belonging to a large family of computational architectures, that differ on various dimensions of form or function, but possess a general family resemblance. Recall the wide variety of "robots" depicted in the Star Wars trilogy? Similarly, we can imagine a wide variety of WebBot-like "digital analogues."

The research lineage of approaches as ours can be traced to several sources, with perhaps the general theme mostly related to Negroponte (1970), though the concept of an "intelligent agent" has been around in thought, though not in form, since at least the 1960's. Several fascinating perspectives are abound in the field, such as knowbots (Kahn & Cerf 1988), softbots (Etzioni, Lesh &

Segal 1994), varieties of software agents (Genesereth & Ketchpel 1994; Geif 1994, Guha & Lenat 1994; Keller 1994), apprentices (Dent, Boticario, McDermott, Mitchell & Zabowski 1992), intelligent agents (King 1995; Roesler & Hawkins 1994), distributed intelligent agents (Hayes-Roth 1990; Rosenschein 1992), and a host of similar creatures in the distributed artificial intelligence literature (e.g., Gasser & Huhns 1989).

The options for organizations currently range from purchasing available application specific software (e.g., generally for information retrieval, data mining, or news filtering), building their own agents within a particular technology using a form of scripting language, such as General Magic-like Telescripts (White 1994), crafting their own proprietary systems for specific purposes, or hiring a firm to build or apply agent technology (e.g., Comshare or Andersen Consulting's Enterprise Intelligent Systems group). Additionally, research projects are underway to provide general agent design languages and open architectures (e.g., Cohen et al. 1994; Shoham 1993). However, one most be careful to understand the "granularity and form" of the architectures and languages. For example, there are large differences between building agents from enhanced components of a programming facility, like a pre-defined object package within C++, and building agents from a much higher architectural level, such as those often afforded by the distributed artificial intelligence approaches (Bond & Gasser 1988). One goal of this chapter is to add an approach to this last list of efforts that brings a quite different perspective on agent design.

Our collective role in this chapter is not one of historian; consequently, we are permitted to exploit the available degrees of freedom afforded by this claim to offer our own interpretation and work from there. Our first interpretation is as follows:

> A *WebBot* is a computer program that operates autonomously to accomplish a task or set of tasks as an intellectual advisor and assistant to a human counterpart.

WebBots of the sort we are describing, then, are presented with goals and "turned loose" within a system or a network (or within many networks) to accomplish some electronic type tasks. We might tell a *WebBot* (let us ignore issues of natural language communication) to:

- monitor who logs in a set of terminals and report on Monday the list of people who... [a network monitoring *WebBot*]
- check the Yahoo reference every day for new additions to telecommunications home pages that... [an Internet watcher WebBot] and then add them to a resource list... [an Internet fetch WebBot]
- keep an up-to-date list on the references of recent hearings on the cable

industry where... [an Internet fetch WebBot] and then go get the text if it is on-line... [an Internet fetch WebBot]

- watch the corporate knowledge bases for any new additions regarding audit and technology issues in the health care industry... [a corporate watcher WebBot]... and send them email requesting a copy of their knowledge report where... [a corporate communicator WebBot]
- go out and scan the employee's disks checking for viruses and report back the results... [a virus checking WebBot]

In the first example, a network administrator might want to check idle time and workloads for a subset of the terminals. For the second, a WebBot will access an Internet search resource (Yahoo) every day to determine if new home pages (i.e., World Wide Web sites) have been added for a particular topic. The third example, addresses what might have been called "library research" where the electronic card catalogs are periodically monitored and text resources are obtained when appropriate. The fourth example depicts a corporation that has set up a type of knowledge base containing experiences (e.g., problems, solutions, and explanations) of client engagements, or internal projects. The last example illustrates a WebBot who has virus monitoring responsibility for software on employee's disks.

The set of WebBots available to their human counterparts are defined in terms of the *business processes they are to accomplish* at "one end" with the embedded procedural knowledge of how to accomplish them at "the other." The embedded procedural knowledge, as one may surmise, must include aspects of the digital image of the relevant corporate resources that enable it to perform the tasks at hand. WebBots must know about (or have the capacity to figure out) the corporate environment within which they reside.

Four general observations can be made on the requirements of the WebBots we are describing that differentiate them from the fellow agents. First, if an organization (we will assume that organizations may be some of the major investors in such creatures) engages a set of WebBots as assistants for their employees, and employees in organizations often require interaction with other employees, then an obvious observation can be made:

> Some of WebBots' tasks will necessitate interacting with other WebBots.

Second, business processes themselves can be rather complex and, in fact, constructed (defined in terms) of simpler business processes. That is, business processes will be specified in terms of other business processes. Therefore, a second observation can be made:

> Some of the WebBots' tasks will consist of sequences of, or even

hierarchically defined levels, of business processes.

Third, the WebBots must be able to engage a sufficient amount of deliberation to reasonably deal with the vague, ambiguous, and uncertain environment encountered in the attempted execution of the business processes within a dynamic corporate setting. That is:

WebBots should have the capacity to reason about, and learn from, their actions.

Finally, there is a direct implication of two types of potential communication in this simple structure: human-WebBot, WebBot-WebBot. Human-WebBot communication is, essentially, a fundamental question of human-computer interface development. This is a critical element in systems design, as from the perspective of the user, the interface *is* the system. Good systems design teams understand and address this issue. Tell the WebBot what to do (e.g., via some scripting language, mouse clicks, voice commands), and have the WebBot report back to the user in the most appropriate (or desired) manner available.

What is interesting are the implications of the WebBot-WebBot communications. What might they say to each other? How should they say it? In part, these are also system design questions whose answers depend on the nature of the problems to be addressed, and by the particular agent technology. One could imagine that this would involve communication for requesting and providing information for task-related purposes, including coordination. Yet, we are envisioning communication also occurring on a fundamentally different level. Because of the particular architecture out of which our WebBots are constructed, they are quite capable of direct knowledge exchange (DKE). The fundamental structures that compose their learned knowledge can be shared among WebBot agents. (Note that agents do not need to be actually "built out of" this particular architecture; rather, they simply have to include a component of their architecture that is DKE-enabled.) DKE-enabled WebBots have a remarkable capability:

WebBots can directly share the knowledge they accumulate in the performance of their tasks.

We are describing WebBots that have a fundamental *intelligence* and that are able to reason about their task and their environment, which includes the behavior of other WebBots. Furthermore, each WebBot is capable of educating any other WebBot by directly communicating its knowledge. There could be, of course, much simpler and specialized forms of WebBots, but in this chapter we are addressing a more ambitious species – a species capable of rudimentary problem solving and learning (thus changing its behavior) in the service of a goal.

Thus we can offer a modestly improved definition:

> A *WebBot* is a computer program that operates autonomously and intelligently to accomplish a task or set of tasks as an intellectual advisor and assistant to a human counterpart and other WebBots.

A Computational Study of WebBots

As we are also envisioning many WebBots interacting with each other, we are also describing an embedded electronic population of WebBots interacting within an explicit (or implicit) WebBot organizational structure. On one level, as defined, there is a distinct set of business processes that are defined in terms of the WebBots which instantiate them. On another level, there is a new and different organization of agents interacting in a world influenced, but not populated by, humans. Our primary interest in WebBots is not in what they might be able to achieve, but rather in the implications of WebBot social interactions with each other in this electronic organizational sub-culture.

What can we begin to say about an organizational sub-structure of intelligent WebBot agents? We conducted a preliminary computational exploration of a simple organizational situation based on the type of WebBot we have defined. We asked the following question: to what extent does WebBot honesty affect individual and collective organizational behavior?

Organizations are comprised of individual agents whose collective activity defines the "behavior" of an organization. Similarly, the individual decisions of WebBots affect the behavior of the entire WebBot collective and, consequently, the organizations general behavior. From this perspective, certain types of WebBot's social behaviors (i.e., interacting with each other within the organization in the execution of their tasks) and their effects on organizational performance (individual and collective properties of their behavior) are explored. The WebBot's social behavior are defined by a set of behavioral predispositions they have reflecting specific "rules of social engagement" defining a rudimentary social cognition component of WebBot deliberation. The rules we are investigating are concerned with honesty and benevolence judgments within the context of a socially-situated task.

The simulation described in this chapter is unique. It reflects a "theory upon a theory" as the fundamental WebBot architecture is itself a theory of individual intelligence, called *Soar*. With this theory of individual WebBot architecture forming the basis for creating WebBots, an assemblage of WebBots are linked together, interacting in a social environment in their performance of a task.

Two types of knowledge are encoded in each WebBot: task-specific knowledge, enabling the task to be accomplished as well as social-interaction knowledge, reflecting the social cognition rules of social engagement for the properties investigated (honesty, benevolence). By situating these WebBots in an organizational task that permits social interaction, a small organizational unit is defined. How each WebBot behaves is based on the nature of the goals, the knowledge available to be brought to bear to work on those goals, and how the task unfolds in the context of other WebBots.

In cognitive science terms, each WebBot defines its own *problem space* reflecting critically perceived aspects of the task environment (Newell & Simon 1972). Each WebBot's problem space also contains models of other encountered WebBots and their behaviors, for these other WebBots are also components of the task environment. It is from models of each other's behaviors that decisions are made regarding interactions, and it is the nature of these social interactions (i.e., interaction decisions) which define collective organizational behavior. Yet, each WebBot constructs its task-specific social reality, and performs its problem solving behaviors, in the same manner and with the same underlying architecture. It is entirely knowledge-based, with a single set of mechanisms operating under a unified approach to defining all aspects of deliberate problem solving – Soar.

The Soar Architecture

Soar is a symbol-oriented computational architecture for general intelligence (Laird, Newell, & Rosenbloom 1987; Newell 1990). In the Soar architecture, tasks are represented as *search*, via the application of *operators* to manipulate symbol structures in *working memory*, within *problem spaces* to achieve *goals* (a particular symbol structure). Knowledge in Soar is represented as if-then productions. If Soar cannot directly and unambiguously achieve a goal with its current knowledge base, the architecture automatically generates an *impasse*, causing a new goal, called a subgoal, to be created and addressed – the resolution of the impasse. This, in turn, may cause further impasses.

Essentially using a depth-first, look-ahead search, the *subgoaling* process proposes "hypothetical" *sub-problem spaces* that correspond to each of the available actions. Soar traces the decision trees that would unfold if each possible *sub-problem space* was in fact chosen, and then evaluate the outcomes of each alternative action in terms of the current goal. Since both the production memory and the working memory are always accessible and the same problem solving mechanisms apply in any *sub-problem space*, the full problem solving power of Soar is available for each sub-problem.

Once Soar resolves a subgoal, an analysis of the working memory elements (symbol structures) leading to the resolution of the subgoal is made, linking them to the eventual working memory elements of the resolution. From this, new long-term productions are created, called chunks, which represent accumulated knowledge to directly resolve the state causing the impasse if it is again encountered, thus avoiding subgoal deliberations. Soar has learned.

All decisions in Soar are made in a two phase decision cycle. During the first phase, called the elaboration phase, any and all productions whose left-hand conditions have been satisfied fire, adding their contribution to the working memory. As working memory changes, different productions may become able to fire, while others may lose support. This process continues until no more productions may fire or an arbitrarily large (essentially infinite) number of passes through the production memory have been made. Thus, all productions effectively fire in parallel. The decision cycle in Soar represents the fundamental metric for deliberation: more cognitive cycles, more cognitive effort.

Soar-based WebBots

With much of the fundamental effort of intelligent deliberation a component of the architecture, WebBots can be created from Soar by adding task-specific knowledge. The task for the study consisted of having WebBots search out electronic information resources over a network. A WebBot would receive a net resource to find (e.g., a Web site that contains some desirable information resources), then proceed to search for the Web site containing resource. An option available to the WebBot is to send out an electronic message to other WebBots to see if they have encountered this requested resource and where it might be located.

In modeling the access to the various resources, we imposed a sequence of Web-based processes required to access them. For example, one might imagine logging in to a mailbox to get an assignment. send out the email, getting a response, evaluating the response, logging in and accessing/searching through a variety of interim Web pages or pointers to finally access the resource, which may or may not be at a particular Web site..

The WebBots themselves were provided with the following rudimentary characteristics:

- *communication* – can ask other WebBots if they have seen a resource on the net, can answer other WebBots' questions regarding the same issue;
- *location memory* – can recall what resources it has seen when it has visited a net site;
- *social memory* - can recall its interactions with other WebBots regarding

requests for net site information (i.e., was it correct or not);

- *rules of social engagement* – when asked for information, would the WebBot consistently tell the truth (trustworthy WebBots), or would it consistently mislead (untrustworthy WebBots);
- *social judgment* – a scoring scheme for judging whether a WebBot was trustworthy based on social memory of past communications and engagements.

For this study, five different organizational sizes were examined (one through five WebBots), and each organization was homogeneous with respect to one of two conditions. In the first condition, the set of WebBots were all honest – they attempted to respond accurately to questions of possible Web resource locations from either their memory or from their current perspective (i.e., directly observing it from their location). In the second condition, all WebBots attempted to deceive other WebBots when they received requests for Web locations.

In both conditions all WebBots engaged in social judgments. Social judgment consisting of ratings of trustworthiness based on the veracity (or lack thereof) of each of the other WebBots' past communications. WebBots incorporate three levels of trustworthiness of an information source: trustworthy (location of a Web site was correct), possibly untrustworthy (location of last requested Web site was incorrect), and untrustworthy (location of last two requested Web sites were incorrect). Thus, if at some time in the past WebBot X told WebBot Z that net resource A was at net site ξ, and in acting on this information, WebBot Z finds that net resource A is indeed at net site ξ, then WebBot Z's "opinion" of WebBot X would support a social judgment of trustworthiness. If WebBot Z's prior opinion of WebBot X was "possibly untrustworthy," then that opinion would be upgraded to trustworthy.

On the other hand, if WebBot Z fails to find net resource A at net site ξ, then WebBot Z would downgrade its opinion of WebBot X. If WebBot X was previously considered as "trustworthy" (the initial judgment values of all WebBots), then it would be downgraded to "possibly untrustworthy." Two consecutive incorrect messages from a given WebBot results in a judgment of "untrustworthy." Once some WebBot Z deems another WebBot X as untrustworthy, WebBot Z automatically presumes that all further communication from WebBot X will be also incorrect. These WebBots are not forgiving.

The characterizations of WebBots are highly stylized, but represent the facets of a broad range of behaviors found in functional and dysfunctional human agents (or human-created agents, like computer viruses and cracker codes). As WebBots can be programmed to perform in any particular manner, and as organizational members may compete, in part, in terms of the behaviors of their personal or corporate WebBots, this represents a legitimate investigation of extreme, though possible, (non)cooperative behaviors. The primary point in this chapter is to illustrate a methodology, present a small example of a computational study, and report its results.

The net environment was depicted as having fifteen single net resource orders for each WebBot, and twenty possible net locations with three potentially relevant resources at each location. There are no duplicates in either the task orders or the available resources; therefore, there are forty-five additional extraneous resources distributed throughout the net locations (accounting for occasional additional local search effort). Each WebBot would request one net task order at a time, locate the resource in the net environment, then request another net task. The simulations were run until the order queue was completed. The simulations were conducted on networked workstations, each running a single copy of a Soar WebBot agent.

The Results

Several dependent measures were used to examine the behavior of the individual and collective WebBots. "Cognitive effort" is a general metric based on decision cycles. *Average cognitive effort* is the total cognitive effort (in terms of decision cycles) divided by the number of WebBots in the organization. Figure 1 presents the data for all five organizations, and for both organizational types. As can be seen in Figure 1, the more WebBots in the organization, the less each WebBot has to do. Yet, except after the first precipitous drop (from one to two WebBots), the additional reduction in load does not seem that remarkable.

Insert Figure 1 about here.

Maximum cognitive effort reflects the most cognitive effort done by a WebBot in an organization. By making a simplifying assumption that all decision cycle efforts are equivalent in duration, this is also a measure of the *total time* required to complete the task. From the figure it is clear that some events are effecting the two organizations differently. The organization comprised of Untrustworthy agents first declines in terms of total time taken to complete the task (with organizations consisting of one, two, and three WebBots), then begins to rise (with organizations of four and five WebBots). Organizations with Trustworthy agents have a similar profile, but its "periodicity" is smaller, where the decline/increase cycle occurs twice, with local minima at two WebBots and four WebBots respectively.

Insert Figure 2 about here.

This graph begs three questions: What is happening to Trustworthy organizations with three WebBots? What is driving the curves up as the size of the organization increases? What is causing the interaction between Trustworthiness and organizational size (i.e., Untrustworthy WebBots dominate at lower organizational sizes, Trustworthy WebBots dominate at large organizational sizes)?

Regarding the first question, the answer is found in the distribution of effort among the WebBots. Essentially, in this organization there was a significant, but anomalous, maldistribution of effort centered around one WebBot that accounted for over 53% of the search effort, which was highly unusual (e.g., the distribution for the three Untrustworthy WebBots was 34%, 34%, 31%).

The second question concerns the curves themselves as they eventually rise (i.e., the time to complete the task increases) as the number of WebBots increases beyond four or five because of several interacting events. Contributing to this is a general increase in *wait* time as the number of WebBots begin (slightly) to interfere with each other as they try to access resources. *Wait cycles* indicate the amount of time the WebBot spends simply in queues to access a net resource. This usually represents a wait when multiple WebBots are trying to access a net location, which, in this model, locks out all but one WebBot at a time. In Figure 3, the rate of increase of wait cycles is apparent with the graph of average wait cycles per WebBot.

Insert Figure 3 about here.

Also influencing the general rise in task time is the contribution made by the ability to communicate. Communication, in this model, takes time and effort for a WebBot. Therefore, asking and answering questions represents a general contribution to task time, as shown in Figures 4 and 5 respectively. An examination of Figure 4 yields a general increase in communication effort as the size of the organization increases. This is expected, as more agents are attempting to communicate. Similarly, in Figure 5, more answers are being generated in response to the questions.

Insert Figures 4 and 5 about here.

Thus, both wait time and communication time drive up the overall time to complete the task. But what about the third question? That is, why the interaction between task time and WebBot trustworthiness? In part, this has already been addressed. The differential contributions of both wait cycles (Figure 3) and questions answered (Figure 5) account for this interaction.

It is important to make a distinction between the total *time* it takes an organization to complete the task, and the total *effort* (in terms of aggregate WebBot decision cycles) it takes to complete the task. From prior Figure 1, it was clear that the average WebBot effort was decreasing, and from prior Figure 2 it was clear that the total time was decreasing (then increasing slightly). Figure 6 shows the cumulative organizational effort it takes to complete the task. The total cognitive effort to complete the task steadily *increases* as more WebBots are brought onboard. *Total cognitive effort* is measured as the sum of the number of decision cycles that all WebBots in an organization require to complete a task. Note again the slight interaction between organizational size and WebBot trustworthiness.

Insert Figure 6 about here.

Finally, there seems to be another recurring periodicity occurring in Figure 6 for both types of organizations across sizes, though the "phases" are off. Of interest, then, is the attenuation of the graph for the NonTrustworthy WebBots and the increase of the graph for the Trustworthy WebBots at the larger organizational sizes.

The answer to both questions can be found in an investigate of question/answer events. Figures 7 and 8 show the average number of questions asked (Figure 7) and answered (Figure 8) for the organization sizes.

Insert Figures 7 and 8 about here.

As the number of WebBot agents increase, Trustworthy agents ask and answer more and more questions. The reason for this is that more information is being discovered as the WebBot agents search the Net. The more they search, the more they learn. The more they learn, the more they can communicate.

On the other hand, NonTrustworthy WebBots are actually asking *decreasing* the amount of questions they are asking and answering. As the number of WebBot agents are able to encounter and communicate wrong/unreliable information, specific decisions are made to not ask specific agents. Remember that a WebBots gents only give wrong information about a net resource only if they have encountered that resource. As the organization learns about its environment, it learns not to trust more and more of its WebBots. The penalty incurred for (at first) trusting untrustworthy agents is that a WebBot accesses specific locations (based on information from other WebBots) only to find the desired resource not there (see Figure 9). In this figure Trustworthy WebBot agents are never mislead, and NonTrustworthy WebBot agents begin to incur effort costs as the size of the organization grows.

Insert Figure 9 about here.

Conclusion: About Organizational Science

KATHLEEN: HERE ARE SOME SENTENCES. CAN YOU FIX THIS LAST PART?

It is estimated that by the year 2000, intelligent agent revenues in the United States and Europe could reach $4 billion (Fletcher, 1994).

Raj Reddy of Carnegie Mellon University predicts that you will be able to "buy these agents [as you would] templates for spreadsheets" (Anthes, 1995.

Microsoft is already building the successor to (the unfortunate) Microsoft Bob, called Peedy, an "anthropomorphic 3-D rendered parrot" to serve as an intelligent assistant (Rupley, 1995).

There are already a variety of agents (a.k.a. webcrawlers, robots, spiders) that are used to search, index, and retrieve information on the World Wide Web. Even one that simply (and usefully) measures the growth of the Web (Indermaur, 1995).

Though not addressed in this chapter, there is another capability of our WebBots: to search out skills (as knowledge) on networks. The DKE capability of these types of intelligent agents is critical in evolving the nature of the intelligent agent to a form that is both capable of, and sharing in, the accumulation of knowledge in the performance of organizational tasks. These agents are not simply intelligent explorers, they are intelligent *knowledge explorers*.

We conclude this chapter with speculations about the future of a form of organizational science. An organizational science that is based on conducting computer-based simulations of computer-realized organizations comprised of artificially intelligent agents.

Why can we feel confident in the prediction that they will play an important part of the future of organizations? And why might we argue for the organizational science of WebBots? The answer lies in the very nature of the emerging information technologies, the organizations embracing these technologies, and the businesses environments where the organizations use these technologies. The wave of technological change that is assaulting corporations and lying the groundwork for the opportunity and need for organizational WebBots.

Human agents have a rich effloresce of knowledge and behaviors based on their years of experience and the remarkable capacity of the human brian. However, many types of organizational positions impose a variety of constraints which effectively inhibit much of the reasoning capabilities of the employee. As a consequence, most of the variance of the performance of interest in organizational settings can be accounted for by a relatively small set of recurring response behaviors (perhaps expressed as rules) to the goals of the situation under the particular constraints. Thus, by selecting a small set of response behaviors (in this case, rules of social engagement) occurring in a task situation that is highly constrained (the task is both simple and restricted in execution), we can model those computationally. By systematically manipulating the social "rules of engagement" knowledge (and thus social behavior) of individual agents, we can systematically observe the emerging organizational behavior, for organizational behavior is directly derivative from the nature of the individual agents and the context within which they interact. This chapter, then, is an incremental step in exploring aspects of ACTS theory and computational modeling of organizations.

References

Anthes, G. Great Expectations, *Computerwold*, April 3, 1995, 89-90.

Bond, A. & Gasser, L. (Eds.). *Readings in Distributed Artificial Intelligence*, San Mateo, CA: Morgan Kaufmann, 1988.

Cohen, P., Cheyer, A., Wang, M. & Baeg, S.C. An open agent architecture. *Paper* *presented at the 1994 AAAI Spring Symposium on Software Agents*, 1994.

Dent, L., Boticario, J., McDermott, J., Mitchell, T. & Zabowski, D. A persoal learning apprentice. *Proceedings of the Tenth National Conference on Artificial Intelligence.* AAAI Press/MIT Press, 1992, 96-103.

Etzioni, O., Lesh, N. & Segal, R. Building Softbots for Unix. In O. Etzioni (Ed.), *Software Agents: Papers from the 1994 Spring Symposium,* AAAI Press., 1994.

Fletcher, P. Intelligent agents will have huge impact on software. *Electronics*, August 8, 1994, 2.

Gasser, L. & Huhns, M. N. (Eds). 1989. *Distributed Artificial Intelligence*. San Mateo, CA: Morgan Kaufmann.

Greif, I. Desktop agents in group-enables products. *Communications of the ACM*, 37(7), 1994, 100-105.

Genesereth, M. & Ketchpel, S. Software agents. *Communications of the ACM*, 37(7), 1994, 48-53.

Guha, R. & Lenat, D. Enabling agents to work together. *Communications of the ACM*, 37(7), 1994, 127-142.

Hayes-Roth, B. Architectural foundations for real-time performance in intelligent agents. *The Journal of Real-Time Systems*, 2, 1990, 99-125.

Houlder, V. Special Agents, *Financial Times*, August 16, 1994, 12.

Indermaur, K. Baby Steps. *Byte*, March 1995, 97-104.

Kahn, R. & Cerf, V. *An Open Architecture for a Digital Library System and a Plan for its Development*. Corporation for National Research Initiatives, Technical Report, 1988.

Keller, J. Info-Age Slaves, *Wall Street Journal*, November 14, 1994, col 1.

King. J. Intelligent Agents: Bringing Good Things to Life. *AI Expert*, Februrary, 1995, 17-19.

Laird, J., Newell, A. & Rosenbloom, P. Soar: An Architecture for General Intelligence. *Artificial Intelligence*, 33(1), 1987, 1-64.

Negroponte, N. *The Architecture Machine: Towards a More Human Environment*. Cambridge, MA: MIT Press, 1970.

Newell, A. *Unified Theories of Cognition*. Cambridge, MA: Harvard 1990

Newell, A. & Simon, H. *Human Problem Solving*. Englewood Cliffs, NJ: Prentice-Hall, 1972.

Roesler, M. & Hawkins, D. Intelligent agents. *Online*, July, 1994, 19-32.

Rosenschein, S. J. 1992. Distributed Intelligent Agents: Person Computer and Intelligent Systems. *IFIP 12th World Computer Congress Proceedings. IFIP Transactions* A-14, 61-63.

Rupley, S. Intellient agents meet the social interface. *PC Magazine*, March 1995, 32.

Shoham, Y. Agent-oriented Programming. *Artificial Intelligence,* 60, 1993, 51-92.

White, J. *Telescript technology: The foundation for the electronic marketplace.* White Paper, General Magic, Inc. Mountain View: CA, 1994.

Acknowledgments

We would like to thank our colleagues.