Collaborative Decision-Support and the Human-Machine Relationship

Jens Pohl, Executive Director
Collaborative Agent Design (CAD) Research Center
Cal Poly University, San Luis Obispo, California, USA

Some Underlying Human Realities.

Human beings are inquisitive creatures who seek explanations for all that they observe and experience in their living environment. While this quest for understanding is central to our success in adapting to a changing environment, it is also a major cause of our willingness to accept partial understandings and superficial explanations when the degree of complexity of the problem situation confounds our current cognitive capabilities. In other words, a superficial or partial explanation is considered better than no explanation at all. As flawed as this approach may be, it has helped us to solve difficult problems in stages. By first oversimplifying a problem we are able to develop an initial solution that is later refined as a better understanding of the nature of the problem evolves.

Unfortunately, now we have to contend with another characteristic of human beings, our inherent resistance to change and aversion to risk taking. Once we have found an apparently reasonable and workable explanation or solution we tend to lose interest in pursuing its intrinsic shortcomings and increasingly believe in its validity. Whether driven by complacency or lack of confidence, this state of affairs leads to many surprises. We are continuously discovering that what we believed to be true is only partly true or not true at all, because the problem is more complicated than we had previously assumed.

At times a particular set of explanations, or school of thought, becomes entrenched as a paradigm that is not easily broken. Kuhn (1977) has drawn attention to the stagnating influence on progress of scientific paradigms, the resistance experienced by individuals or small groups that wish to correct flaws in a paradigm, and the resurgence of innovative activity after the paradigm has been broken. If experts in science will succumb to this weakness in human nature then how much more difficult will it be for a layperson to maintain a discerning mind?

Throughout modern history these intrinsic human characteristics of resisting change, avoiding risks, and endeavoring to maintain status quo have created a tension in society. A prominent example is, of course, the Information Revolution driven by the rapid development of computers and communication systems and their potential assistance in human decision making endeavors.

The Increasing Complexity of Problems in a Global Community.

The complexity of problems faced by human society in areas such as management, economics, marketing, engineering design, military operations, and environmental preservation, is increasing for several reasons. First, computer-driven information systems have expanded these areas from a local to an increasingly global focus. Even small manufacturers are no longer confined to a regionally localized market for selling their products. The marketing decisions that they have to

make must take into account a wide range of factors (e.g., international currency rates, political alliances, and climatic conditions) and a great deal of knowledge (e.g., language, conventions, and cultural beliefs) that is far removed from the local environment.

Second, as the scope of the problem system increases so do the relationships among the various factors. These relationships are difficult to deal with, because they require the decision maker to consider many factors concurrently. Although the biological operation of the human brain is massively parallel, our conscious reasoning processes are sequential. Simply stated, we have difficulty reasoning about more than two or three variables at any one time.

Third, as the scope of problems increases decision makers suffer simultaneously from two diametrically opposed but related conditions. They tend to be overwhelmed by the shear volume of information that they have to consider, and yet they lack information in many specific areas. To make matters worse, the information tends to change dynamically in largely unpredictable ways.

It is therefore not surprising that governments, corporations, businesses, down to the individual person, are increasingly looking to computer-based decision-support systems for assistance. This has placed a great deal of pressure on software developers to rapidly produce applications that will overcome the apparent failings of the human decision maker. While the expectations have been very high, the delivery has been much more modest. The expectations were simply unrealistic.

It was assumed that advances in technology will be simultaneously accompanied by an understanding of how these advances should be applied optimally to assist human endeavors. History suggests that such an a priori assumption is not justified. There have been countless experiences in the past that would suggest the contrary. For example, the invention of new materials (e.g., plastics) have inevitably been followed by a period of misuse. Whether based on a misunderstanding or lack of knowledge of its intrinsic properties, the new material was typically initially applied in a manner that emulated the material(s) it replaced. In other words, it took some time for the users of the new material to break away from the existing paradigm. A similar situation currently exists in the area of computer-based decision-support systems.

The Rationalistic Problem Solving Tradition.

To understand current trends in the evolution of progressively more sophisticated decision-support systems it is important to briefly review the foundations of problem solving methodology from an historical perspective. Epistemology is the study or theory of the origin, nature, methods and limits of knowledge. The dominant epistemology of Western Society has been technical rationalism (i.e., the systematic application of scientific principles to the definition and solution of problems).

The rationalistic approach to a problem situation is to proceed in well defined and largely sequential steps (Fig.1): define the problem; establish general rules that describe the relationships that exist in the problem system; apply the rules to develop a solution; test the validity of the solution; and, repeat all steps until an acceptable solution has been found. This simple view of problem solving suggested a model of sequential decision making that has retained a dominant position to the present day. With the advent of computers it was readily embraced by 1st Wave

software (Fig.2) because of the ease with which it could be translated into packaged, automated solutions utilizing the procedural computer languages that were available at the time (Pohl 1996). 1st Wave software assumes that problem solving is essentially a sequential process in which every subsequent step depends on the completion of the preceding step. This view of problem solving is far removed from real world experience, where project teams solve problems collaboratively and contribute to the decision making process whenever they have something useful to share with the other team members. Seldom, if ever, is a team member prevented from contributing information until a certain stage or milestone has been reached. On the contrary, team members are encouraged to exchange information freely in the hope that their contributions will accelerate the solution process and increase the quality of the solution.

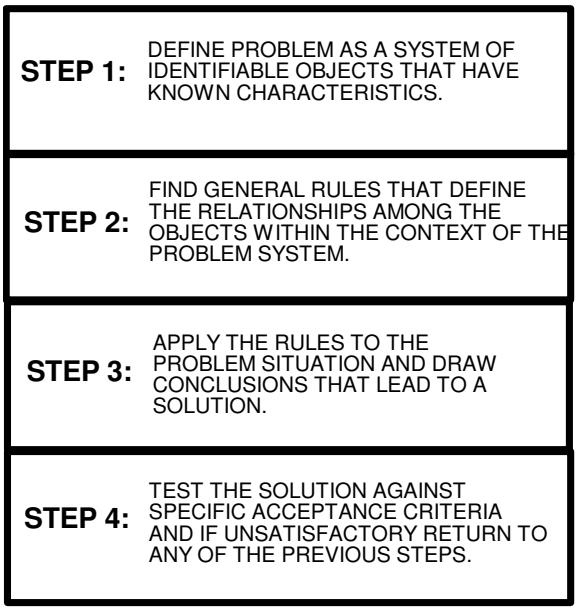

Fig.1: Solution of Simple Problems

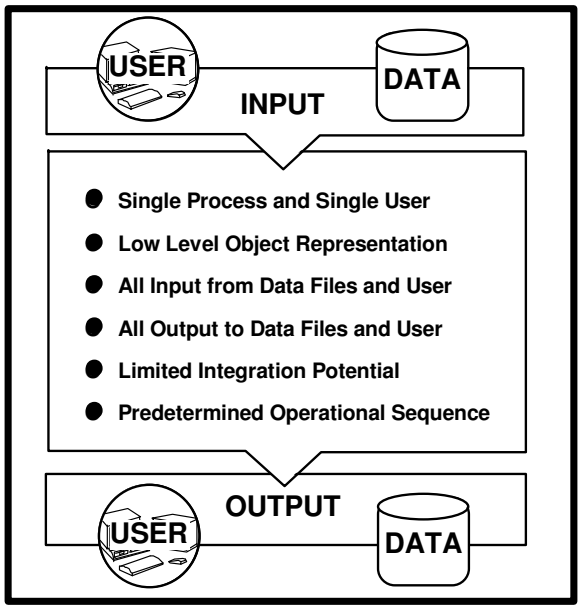

Fig.2: '1st Wave' Computer Applications

Over the past 50 years with the availability of more and more convenient and effective communication facilities, government and industry have been increasingly challenged by real world problems that are often very complex involving many related variables. Neither the relationships among the variables nor the variables themselves are normally sufficiently well understood to provide the basis for clear and comprehensive definitions. In other words, problem situations are often too complex to be amenable to an entirely logical and predefined solution approach. Under these conditions the analytical strategy has been to decompose the whole into component parts, as follows:

- Decompose the problem system into sub-problems.
- Study each sub-problem in relative isolation, using the rationalistic approach (Fig.1), and if the relationships within the sub-problem domain cannot be clearly defined then decompose the sub-problem further.
- Combine the solutions of the sub-problems into a solution of the whole.

Underlying this problem solving strategy is the implicit assumption that an understanding of parts leads to an understanding of the whole. Under certain conditions this assumption may be

valid. However, in many complex problem situations the parts are tightly coupled so that the behavior of the whole depends on the interactions among the parts rather than the internal characteristics of the parts themselves (Bohm 1983, Senge 1993). An analogy can be drawn with the behavior of ants. Each ant has only primitive skills, such as the ability to interpret the scent of another ant and the instinctive drive to search for food, but little if any notion of the purpose or objectives of the ant colony as a whole. Therefore, an understanding of the behavior of an individual ant does not necessarily lead to an understanding of the community behavior of the ant colony of which the ant is a part.

Decomposition is a natural extension of the scientific approach to problem solving and has become an integral and essential component of rationalistic methodologies. Nevertheless, it has serious limitations. First, the behavior of the whole usually depends more on the interactions of its parts and less on the intrinsic behavior of each part. Second, the whole is typically a part of a greater whole and to understand the former we have to also understand how it interacts with the greater whole. Third, the definition of what constitutes a part is subject to viewpoint and purpose, and not intrinsic in the nature of the whole. For example, from one perspective a coffee maker may be considered to comprise a bowl, a hotplate, and a percolator. From another perspective it consists of electrical and constructional components, and so on.

Rationalism and decomposition are certainly useful decision making tools in complex problem situations. However, care must be taken in their application. At the outset it must be recognized that the reflective sense (Schön 1983) and the intuitive capabilities of the decision maker are at least equally important tools. Second, decomposition must be practiced with restraint so that the complexity of the interactions among parts is not overshadowed by the much simpler behavior of each of the individual parts. Third, it must be understood that the definition of the parts is largely dependent on the objectives and knowledge about the problem that is currently available to the decision maker. Even relatively minor discoveries about the greater whole, of which the given problem situation forms a part, are likely to have significant impact on the purpose and the objectives of the problem situation itself.

Decision Making in Complex Problem Situations.

In several previous CAD Research Center publications we have drawn attention to the importance of internal and external relationships in complex problem situations (Pohl et al. 1997 (48-62), Pohl and Myers 1994). As shown in Fig.3, there are several characteristics that distinguish a complex problem from a simple problem. First, the problem is likely to involve many related issues or variables. As discussed earlier the relationships among the variables often have more bearing on the problem situation than the variables themselves. Under such tightly coupled conditions it is usually not particularly helpful, and may even be misleading, to consider issues in isolation. Second, to confound matters some of the variables may be only partially defined and some may yet to be discovered. In any case, not all of the information that is required for formulating and evaluating alternatives is available. Decisions have to be made on the basis of incomplete information.

Third, complex problem situations are pervaded with dynamic information changes. These changes are related not only to the nature of an individual issue, but also to the context of the problem situation. For example, a change in location of an enemy force (even within the same sector of the battlefield) could easily have a major impact on the entire nature of the combat

situation facing the commander. Apart from the disposition of friendly forces under these changed conditions, the influence on target priorities, and the effectiveness of available weapons, such a relocation could call into question the very feasibility of the existing battle plan. Even under less critical conditions it is not uncommon for the solution objectives to change several times during the decision making process. This fourth characteristic of complex problem situations is of particular interest. It exemplifies the tight coupling that can exist among certain problem issues, and the degree to which decision makers must be willing to accommodate fundamental changes in the information that drives the problem situation.

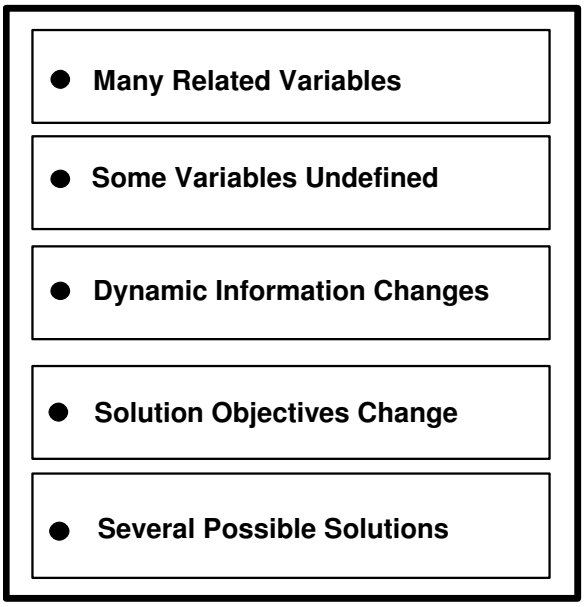

Fig.3: Character of Complex Problems

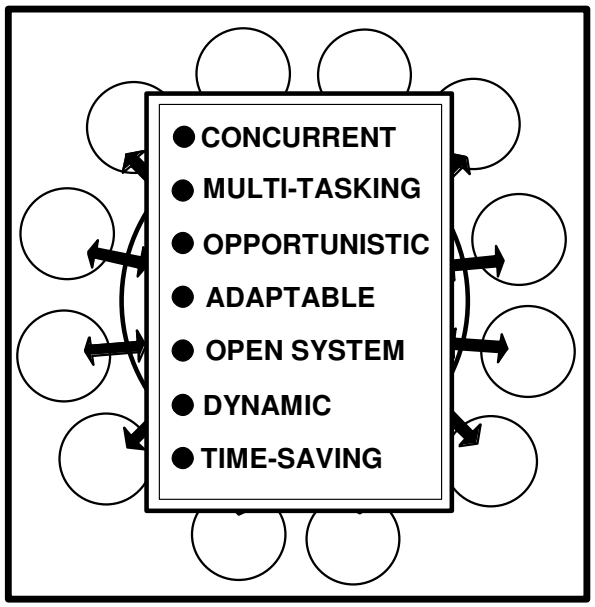

Fig.4: Parallel Decision Support

Fifth, complex problems typically have more than one solution (Archea 1987). It is normally unproductive to look for an optimum solution, because there are no static benchmarks available for evaluating optimality. A solution is found to be acceptable if it satisfies certain performance requirements and if it has been determined that the search for alternatives is no longer warranted. Such a determination is often the result of resource constraints (e.g., availability of time, penalty of non-action, or financial resources) rather than a high level of satisfaction with the quality of the proposed solution.

While human decision making in complex problem situations has so far defied rigorous scientific explanation, we do have knowledge of at least some of the characteristics of the decision making activity.

- Decision makers typically define the problem situation in terms of issues that are known to impact the desired outcome. The relative importance of these issues and their relationships to each other change dynamically during the decision making process. So also do the boundaries of the problem space and the goals and objectives of the desired outcome. In other words, under these circumstances decision making is an altogether dynamic process in which both the rules that govern the process and the required properties of the end result are subject to

continuous review, refinement and amendment. ***Accordingly, the borderline between planning and execution is blurred by the constant need for replanning.***

- The complexity of the decision making activity does not appear to be due to a high level of difficulty in any one area but the multiple relationships that exist among the many issues that impact the desired outcome. Since a decision in one area will tend to influence several other areas there is a need to consider many factors at the same time. This places a severe burden on the human cognitive system. Although the neurological mechanisms that support conscious thought processes are massively parallel, the conscious operation of these reasoning capabilities is largely sequential. Under these conditions the individual human decision maker is very much in need of assistance. ***The availability of computers would appear to offer welcomed support through parallelism (Fig.4), connectivity, and information access, as long as the human decision makers are able to effectively communicate their assistance needs to the computer.***

- Observation of decision makers in action has drawn attention to the important role played by experience gained in past similar situations, knowledge acquired in the general course of decision making practice, and expertise contributed by persons who have detailed specialist knowledge in particular problem areas (Mackinder and Marvin 1982, Mallen and Goumain 1973). The dominant emphasis on experience is confirmation of another fundamental aspect of the decision making activity. Problem solvers seldomly start from first principles. In most cases, the decision maker intuitively builds on existing solutions from previous situations that are in some way related to the problem under consideration. ***Again, computers should be potentially useful through their ability to store not only vast amounts of data but also higher level information and knowledge. It is not unreasonable to expect knowledge-based computer systems (i.e., software applications) to alert the user to past solutions and suggest how these might relate to the current problem.***

- Finally, there is a distinctly irrational aspect to decision making in complex problem situations. Schön (1983) refers to a *"...reflective conversation with the situation..."*. He argues that decision makers frequently make value judgments for which they cannot rationally account. Yet, these intuitive judgments often result in conclusions that lead to superior solutions. It would appear that such intuitive capabilities are based on a conceptual understanding of the situation, which allows the problem solver to make knowledge associations at a highly abstract level. ***This strongly suggests that a collaborative human-computer partnership is essential. Both must contribute their respective strengths and assist each other to overcome their respective weaknesses. Any attempt to automate the decision making process to the exclusion of the human element is not only likely to be counterproductive, but dangerous as well.***

Based on these characteristics the solution of complex problems can be categorized as an information intensive activity that depends for its success largely on the availability of

information resources and, in particular, the experience and reasoning skills of the decision makers. It follows that the quality of the solutions will vary significantly as a function of the problem solving skills, knowledge, and information resources that can be brought to bear on the solution process. This clearly presents an opportunity for the useful employment of computer-based decision-support systems in which the capabilities of the human decision maker are complemented with knowledgebases, expert agents, and self-activating conflict identification and monitoring capabilities.

The Critical Importance of *Information* Representation in the Computer.

Although technological advances in computer hardware and communication systems have been truly astounding over the past 20 years, the direct utilization of these advances in the area of decision-support has been less than remarkable. The fact is that we are still using computers largely as ***data*** processing devices that perform only the most menial and least intelligent data transmission and manipulation tasks. While computers are performing these tasks with great speed and accuracy, and while they are able to provide connectivity among a virtually unlimited number of access points, the higher level and much more rewarding tasks of analyzing, interpreting and abstracting data as ***information*** and ***knowledge*** is almost entirely left to the human users (Fig.5).

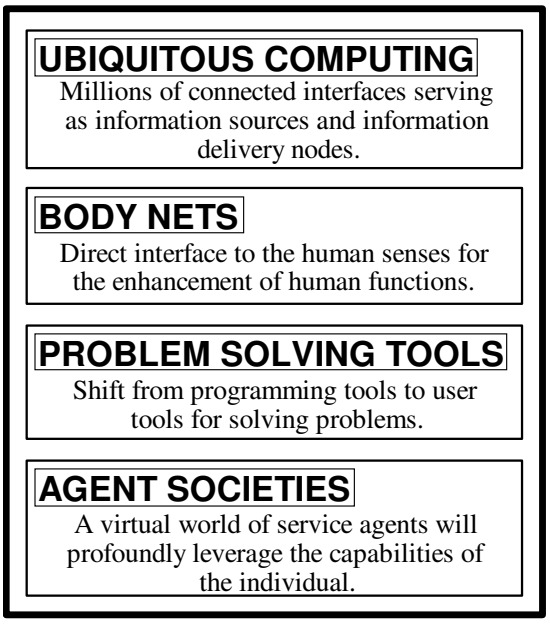

Fig.5: Evolving computer-human partnership

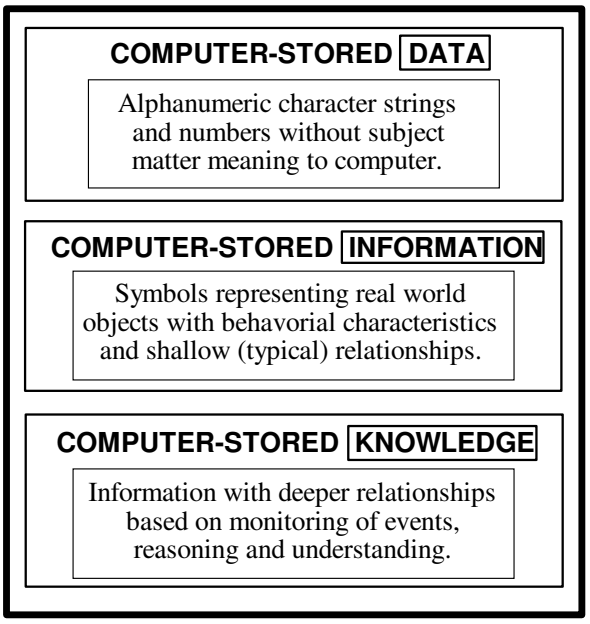

Fig.6: Data-information-knowledge

This serious deficiency has become increasingly apparent as technological advances have increased computing power, data storage capacities, and data transmission speeds by orders of magnitude in such a short period of time. Convenient global access to users and data has increased the need for information filtering, so that individuals might take advantage of the opportunities for material and personal profit that this connectivity and processing power present to the user. Needless to say, the capabilities of a computer to assist in the intelligent assessment of information are basically non-existent if the computer processes this information as bitmaps

and alphanumeric text strings (Fig.6). ***Any significantly useful human-computer collaborative partnership carries with it the expectation that information is held within the system environment in a representational form that is, if not equivalent to, at least compatible with human cognition.***

The current approach for achieving this objective is to represent information in the computer as objects with behavioral characteristics and relationships to other objects (Myers et al. 1993). While this approach is hardly sophisticated it does allow real world objects (e.g., airfield, tunnel, building, weapon, tank) to be represented symbolically so that computer software modules can reason about them.

It is important to note that the relationships among these objects are often far more important than the characteristics that describe the individual behavior of each object. For example, the word *house* holds little meaning if we strip away the many associations that this word represents in our mind. However, such associations to our knowledge of construction materials, our experiences in having lived in houses, and our understanding of how our own home is impacted by external factors (such as rain, sunshine, neighbors, mortgage interest rates, and so on) constitute the rich meaning of the object *house* (Minsky 1982). Accordingly, any useful representation of information in the computer must be capable of capturing the relationships among the entities (i.e., objects) in the problem system.

While some of these associations are fairly static (e.g., a weapon is a kind of asset and a lethal weapon is a kind of weapon) many of the associations are governed by current conditions and are therefore highly dynamic. For example, as a platoon of soldiers moves through the battlefield it continuously establishes new associations (e.g., to windows in buildings from which snipers could fire on individual members of the platoon), changes existing associations (e.g., higher levels of risk as the platoon nears an active combat zone), and severs previous associations (e.g., as the platoon is forced to abandon its compromised command post).

Abstract concepts such as privacy, security and power, are less amenable to this approach since their meaning and role in our day-to-day activities is less easily defined. For example, the characteristics of *privacy* are neither static nor can they be accurately described in relational terms. They depend on a wide range of factors that relate to both environmental and personal circumstances and dispositions. These factors can be only partially accounted for through embedded knowledge and rules, and therefore become largely the purview of the human members of the collaborative human-computer partnership.

Nevertheless, even with these shortcomings this form of representation of real world objects can provide the basis of usable problem solving support and decision making assistance. Improvements are possible with the addition of knowledge bases and user interaction. In the latter case the user becomes as much a helper to the system as the system serves as an assistant to the user. However, this occurs in quite different ways. The system uses its computing and logical reasoning capabilities to monitor, analyze and evaluate the actions, requests and interests of the user in an opportunistic manner. The user, on the other hand, helps the system to understand the nature of the objects and relationships that it is processing in a more deliberate manner (Pohl 1995).

The reliance on object representations in reasoning endeavors is deeply rooted in the innately associative nature of the human cognitive system. Information is stored in long term memory through an indexing system that relies heavily on the forging of association paths. These paths

relate not only information that collectively describes the meaning of symbols such as *helicopter*, *rifle* and *truck*, but also connect one symbol to another. The symbols themselves are not restricted to the representation of physical objects, but also serve as concept builders. They provide a means for grouping and associating large bodies of information under a single conceptual metaphor. In fact, Lakoff and Johnson (1980) argue that *"...our ordinary conceptual system, in terms of which we both think and act, is fundamentally metaphorical in nature..."*. They refer to the influence of various types of metaphorical concepts, such as 'desirable is up' (spatial metaphors) and 'fight inflation' (ontological or human experience metaphors), as the way human beings select and communicate strategies for dealing with every day events. Problem solvers typically intertwine the factually based aspects of objects with the less precise, but implicitly richer language of metaphorical concepts. This leads to the spontaneous linkage of essentially different objects through the process of analogy. In other words, the decision maker recognizes similarities between two or more sub-components of apparently unrelated objects and embarks upon an exploration of the discovered object seeking analogies where they may or may not exist. At times these seemingly frivolous pursuits lead to surprising and useful solutions of the problem at hand.

The need for a high level representation is fundamental to all computer-based decision-support systems. It is an essential prerequisite for embedding artificial intelligence in such systems, and forms the basis of any meaningful communication between user and computer. Without a high level representation facility the abilities of the computer to assist the human decision maker are confined to the performance of menial tasks, such as the automatic retrieval and storage of data or the computation of mathematically defined quantities. While even those tasks may be highly productive they cannot support a partnership in which human users and computer-based systems collaborate in a meaningful and intelligent manner in the solution of complex problems.

The Limited Role of *Visualization*.

Decision makers use various visualization media, such as visual imagination or simulation, drawings and physical models, to communicate the current state of the evolving solution to themselves and to others. For example, drawings, sketches and computer displayed images have become intrinsically associated with problem solving. Although the decision maker can reason about complex problems solely through mental processes, drawings and related visual images are useful and convenient for extending those processes. The failings of a drawing or sketch as a vehicle for communicating the full intent of the decision maker do not apply to the creator of the drawing. To the latter the drawing serves not only as an extension of long term memory, but also as a visual bridge to its associative indexing structure. In this way, every meaningful part of the drawing is linked to related data and deliberation sequences that together provide an effectively integrated and comprehensive representation of the artifact.

From a technical point of view a great deal of headway has been made over the past two decades in the area of computer-based visualization. ***However, without high level representation capabilities even the most sophisticated computer generated images are nothing but hollow shells***. If the computer system does not have even the simplest understanding of the nature of the objects and their associations that are contained in the image then it cannot contribute in any way to the analysis of those objects. On the other hand, visualization in combination with high level representation becomes the most powerful element of the user interface of a decision-support

system. Under these circumstances, visualization promotes the required level of understanding between the user and the computer as they collaborate in the solution of the problem.

The Complementary Role of Human *Intuition*.

Schön (1983 and 1988) has written extensively about the intuitive aspects of decision making. Although he focused primarily on engineering design as an application area, his views provide valuable insight into the solution of complex problems in general. Design has all of the common characteristics of complex problem situations, and some additional ones such as the desire for solution uniqueness, that make it a prime candidate for computer-based assistance (Pohl et al.1994).

In Schön's (1988) view designers enter into *"...design worlds..."* in which they find the objects, rules and prototype knowledge that they apply to the design problem under consideration. The implication is that the designer continuously moves in and out of design worlds that are triggered by internal and external stimuli. While the reasoning process employed by the designer in any particular design world is typically sequential and explicitly logical, the transitions from state to state are governed by deeper physiological and psychological causes. Some of these causes can be explained in terms of associations that the designer perceives between an aspect or element of the current state of the design solution and the prototype knowledge that the designer has accumulated through experience. Others appear to be related to environmental stimuli or emotional states, or interactions of both.

For example, applying Schön's view to the broader area of complex problem solving, a particular aspect of a problem situation may lead to associations in the decision maker's mind that are logically unrelated to the problem under consideration. However, when the decision maker pursues and further develops these associations they sometimes lead to unexpected solutions. Typically, the validity of these solutions becomes apparent only after the fact and not while they are being developed. In popular terms we often refer to these solutions as *creative leaps* and label the author as a brilliant strategist. What we easily forget is that many of these intuitions remain unrelated associations and do not lead to any worthwhile result. Nevertheless, the intuitive aspect of decision making is most important. Even if only a very small percentage of these intuitive associations were to lead to a useful solution, they would still constitute one of the most highly valued decision making resources.

The reasons for this are twofold. First, the time at which the decision maker is most willing to entertain intuitive associations normally coincides with a most difficult stage in the problem solving process. Typically, it occurs when an impasse has been reached and no acceptable solution strategy can be found. Under these conditions intuition may be the only remaining course of action open to the decision maker. The second reason is particularly relevant if there is a strong competitive element present in the problem situation. For example, in command and control situations during the execution of military operations. Under these circumstances, strategies and solutions triggered by intuitive associations will inevitably introduce an element of surprise that is likely to disadvantage the enemy.

The importance of ***intuition*** in decision making would be sufficient reason to insist on the inclusion of the human decision maker as an active participant in any computer-based decision-support system. In designing and developing such systems in the CAD Research Center over the past decade we have come to appreciate the importance of the human-computer partnership

concept, as opposed to automation. Whereas in some of our early systems (e.g., ICADS (Pohl et al. 1988) and AEDOT (Pohl et al. 1992)) we included agents that automatically resolved conflicts, today we are increasingly moving away from automatic conflict resolution to conflict detection and explanation. We believe that even apparently mundane conflict situations should be brought to the attention of the human agent. Although the latter may do nothing more than agree with the solution proposed by the computer-based agents, he or she should be given the opportunity to bring other knowledge to bear on the situation and thereby influence the final determination.

The Human-Computer Partnership

To look upon decision-support systems as partnerships between users and computers, in preference to automation, appears to be a sound approach for at least two reasons. First, the ability of the computer-based components to interact with the user overcomes many of the difficulties, such as representation and the validation of knowledge, that continue to plague the field of machine learning (Thornton 1992, Johnson-Laird 1993).

Second, human and computer capabilities are in many respects complementary (Figs.7 and 8). Human capabilities are particularly strong in areas such as communication, symbolic reasoning, conceptualization, learning, and intuition. We are able to store and adapt experience and quickly grasp the overall picture of even fairly chaotic situations. Our ability to match patterns is applicable not only to visual stimuli but also to abstract concepts and intuitive notions. However, although the biological basis of our cognitive abilities is massively parallel, our conscious reasoning capabilities are essentially sequential. Therefore, human decision makers are easily overwhelmed by large volumes of information and multi-faceted decision contexts. We have great difficulty dealing with more than two or three variables at any one time, if there are multiple relationships present. Under these circumstances we tend to switch from an analysis mode to an intuitive mode in which we have to rely almost entirely on our ability to develop situation awareness through abstraction and conceptualization. While this is our greatest strength it is also potentially our greatest weakness. At this intuitive meta-level we are vulnerable to emotional influences that are an intrinsic part of our human nature and therefore largely beyond our control.

Computer capabilities are strongest in the areas of parallelism, speed and accuracy (Fig.8). Whereas the human being tends to limit the amount of detailed knowledge by continuously abstracting information to a higher level of understanding, the computer excels in its almost unlimited capacity for storing data. While the human being is prone to making minor mistakes in arithmetic and reading, the computer is always accurate. A slight diversion may be sufficient to disrupt our attention to the degree that we incorrectly add or subtract two numbers. However, if the error is large we are likely to notice that something is wrong further downstream due to our ability to apply conceptual checks and balances. The computer, on the other hand, cannot of its own accord distinguish between a minor mistake and a major error. Both are a malfunction of the entirely predictable behavior of its electronic components.

The differences between the human being and the computer are fundamental. All of the capabilities of the digital computer are derived from the simple building blocks of ‘0’ and ‘1’. There is no degree of vagueness here, ‘0’ and ‘1’ are precise digital entities and very different from the massively parallel and largely unpredictable interactions of neurons and synapses that

drive human behavior. It is not intuitively obvious how to create the high level representations of real world objects (e.g., ship, aircraft, dog, house, power, security, etc.) that appear to be a prerequisite for human reasoning and learning, in a digital computer. While these objects can be fairly easily represented in the computer as superficial visual images (in the case of physical objects such as aircraft and house) and data relationships (in the case of conceptual objects such as power and security) that in itself does not ensure that the computer has any understanding of their real world meaning. These representations are simply combinations of the basic digital building blocks that model, at best, the external shell rather than the internal kernel of the object.

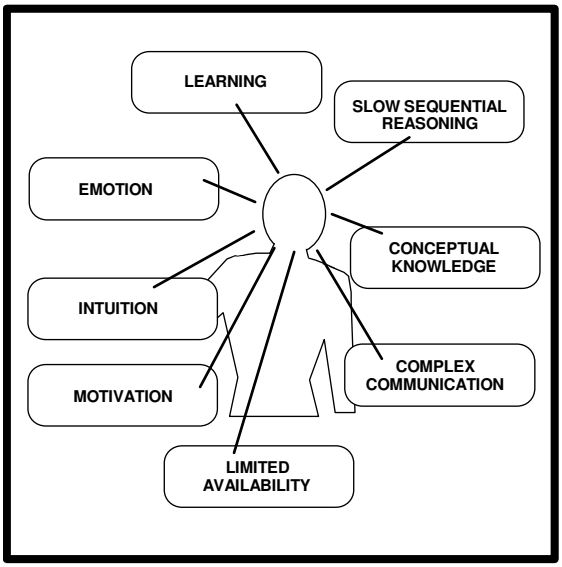

Fig.7: Human Abilities and Limitations

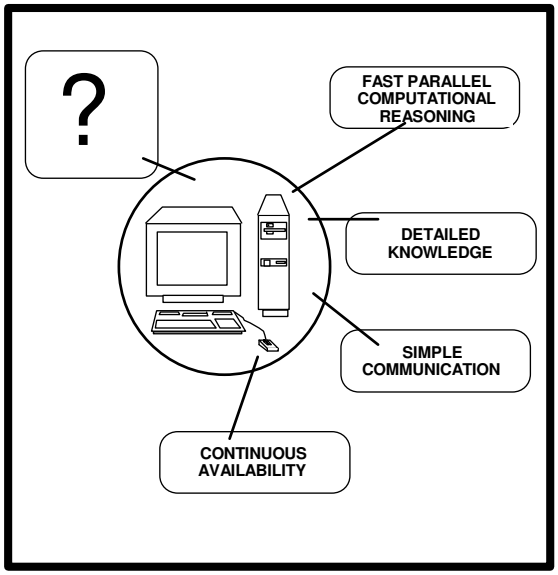

Fig.8: Computer Abilities and Limitations

Unfortunately, it is still not generally understood that this representational inadequacy is the single most limiting factor in virtually all existing decision-support systems. For example, current military command and control systems tend to overwhelm commanders with hundreds of detailed satellite pictures of battlefield conditions that are transmitted by computers as digital packages rather than groups of objects. As a result the interpretation, filtering and fusion of these images, areas in which computer-assistance would be highly desirable, become the burdensome task of the human decision maker.

More than 10 years ago when the CAD Research Center first embarked on the development of cooperative multi-agent systems we recognized the fundamental importance of representation, as a prerequisite for providing computer-based agents with reasoning capabilities. We discovered that while this problem was well known and had been the subject of considerable research in the artificial intelligence community, the results of this research work had generally remained the province of that close-knit community.

Early practical implementations of artificial intelligence systems were almost exclusively confined to stand-alone applications, such as expert systems (e.g., Prospector (Duda et al. 1977, Reboh 1981), MYCIN (Buchanan and Shortliffe 1984), and ASTA (Wilson et al. 1984)). Since these systems were not intended to interface with other applications the importance of representation continued to be largely ignored by the mainstream of software developers and users. Over the past decade the CAD Research Center has explored, adapted and implemented

several high level representation techniques in its various decision-support applications for industry and government sponsors (Myers et al. 1993). While there is a need for a great deal more work in this area the state of technology today is, without question, capable of providing an internal representation level that can support meaningful reasoning assistance in large integrated decision-support systems.

Multi-Agent Collaborative Decision-Support Systems

Adaptation of 1st Wave software (Fig.2) to increasingly more complex real world problem situations has led to a hybrid of human and computer-based decision-support systems (Fig.9). Individual members of the human problem solving team utilize computer-based tools to assist them mostly with the computational and planning components of their tasks. However, this assistance is limited to the individual team member. While the computer can retrieve and send information from and to shared databases, it exercises these capabilities only on the request of its user. Collaboration within the problem team is largely restricted to the communications initiated by team members. The computer shares in these communications only to the extent that its user initiates queries to shared databases. The computer functions as a stand-alone tool that interacts with its user, but does not actively participate in the collaborative problem solving process.

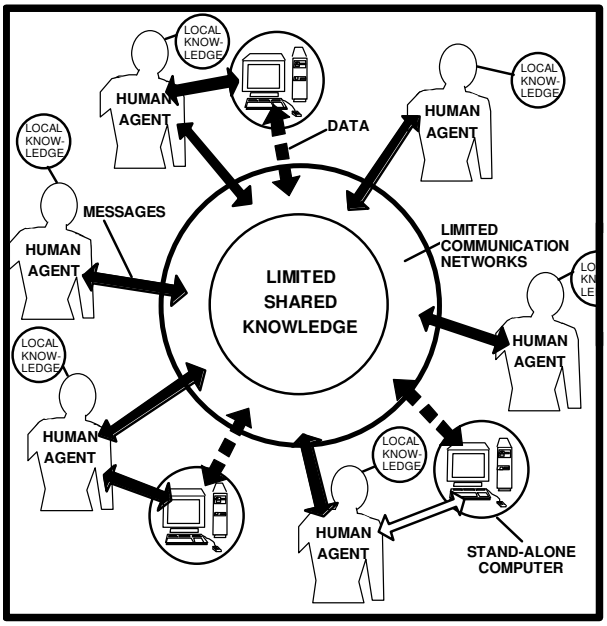

Fig.9: Limited Computer Assistance

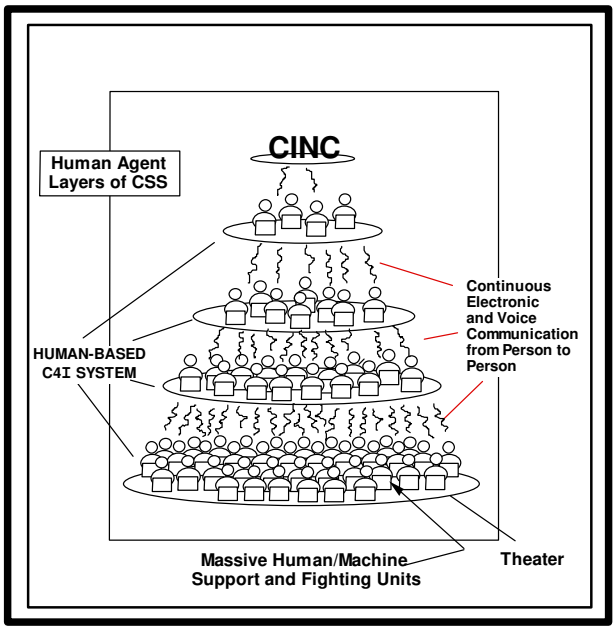

Fig.10: Hierarchical Military C4I Structure

In this hybrid decision-support environment, which is still representative even of the more critical transportation and military systems today, much of the collaboration is based on human to human voice communication. As a result, under severe stress conditions these systems are subject to serious communication bottlenecks that will disrupt and may even terminate the decision making process. In recent years examples of these conditions have occurred during environmental disasters, such as earthquakes in the USA, and military missions, such as Desert Storm in the Middle East. In the latter case, as shown in Fig.10, the combination of a hierarchical command and control structure with a 1st Wave software architecture produced a high potential

for communication failure. A massive build-up of US and allied forces (i.e., more than 500,000 personnel) in the theater was supported by computer-based communication facilities that reflected the chain of command through multiple levels from the commander in chief (CINC) down to the soldier in the battlefield. In this human-based C4I system environment continuous electronic and voice communication, essentially from person to person, quickly clogged the available communication channels.

During the late 1990s the limited computer-assistance capabilities (Fig.9) that are reflective of 1st Wave software will be increasingly replaced by integrated, multi-agent, cooperative systems. This signals the emergence of 2nd Wave software (Fig.11) in which the contributions of several decision-support components are coordinated through an inter-process communication facility. The components, commonly referred to as agents, may be separate processes or modules of one or more processes. They may be rule-based expert systems, procedural programs, neural networks, or even sensing devices. Increasingly, these agents will have the ability to explain their actions and proposals, as they interact spontaneously with each other either directly or through coordination facilities.

In the broadest sense an agent may be described as a computer-based program or module of a program that has communication capabilities to external entities and can perform some useful tasks in at least a semi-autonomous fashion. According to this definition agent software can range from simple, stand-alone, predetermined applications to the most intelligent, integrated, multi-agent decision-support system that advanced technology can produce today.

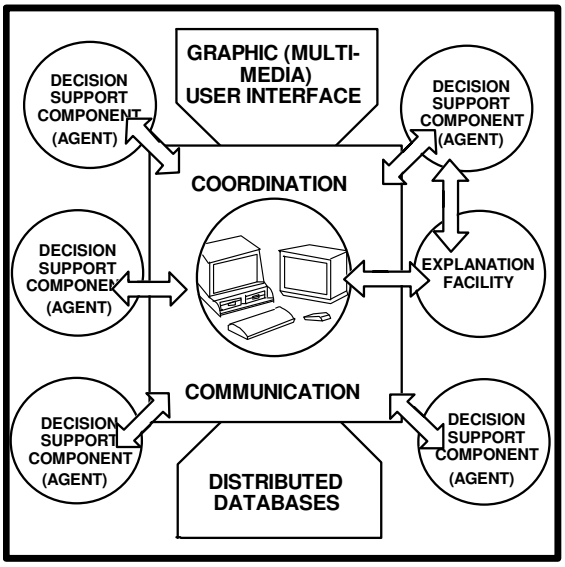

Fig.11: 2nd Wave Computer Applications

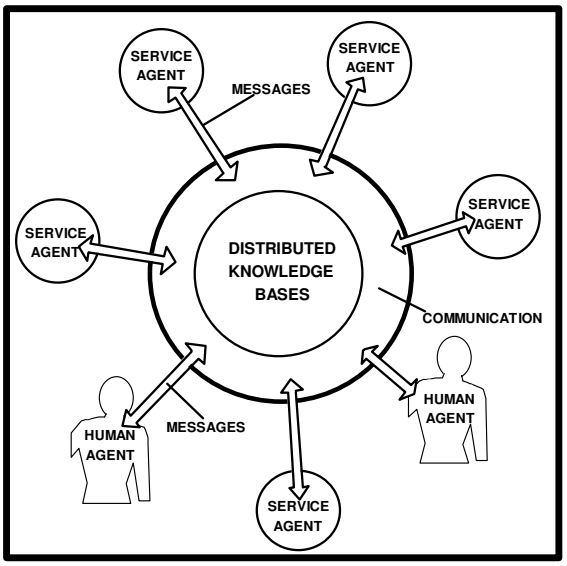

Fig.12: The Service-Agent Architecture

As discussed previously, 2nd Wave software requires a high level internal representation of the real world objects and their relationships that are central to the problem situation. This is a prerequisite for the reasoning capabilities of the agents and also for the interaction of the user(s) with the system. The objective of 2nd Wave software is not to automate the decision making activity, but to create an effective partnership between the human decision maker and the computer-based agents. In this partnership the human agent must be able to communicate with the computer-based agents in terms of the same real world objects that are used so effectively in

all human reasoning endeavors. In their role as active collaborators the computer-based agents will have information needs that cannot be totally predetermined. Therefore, similar to the human agent, they will require the capability to dynamically generate database queries and initiate user interactions. At least some of the information sources accessed by the agents will be prototypical in nature (i.e., standard practices, case studies, and other typical knowledge pertaining to the problem situation) consistent with the notion of knowledge-based systems.

As discussed earlier, human and computer capabilities are complementary in many respects. Where we excel in the areas of abstraction, conceptualization, intuition and creativity, the performance of the computer cannot be described as being even adequate. However, when it comes to computational speed and accuracy, searching for and storing data, redundancy and parallelism, information persistence, and continuous availability, the computer outperforms us by far. It is therefore not surprising that current 2nd Wave software developments are increasingly focusing on collaborative systems in which users interact with computer-based expert agents (Fig.11). Typically, each agent is designed to be knowledgeable in a narrow domain, and represents the viewpoint of that domain in its collaborative endeavors. In this respect it provides services and can be categorized as a service-agent (Fig.12).

The service-agents are endowed with a communication facility that allows them to receive and send information. The manner in which they participate in the decision making activities depends on the nature of the application. They can be designed to respond to changes in the problem state spontaneously, through their ability to monitor information changes and respond opportunistically, or information may be passed to them in some chronological order based on time-stamped events or predefined priorities. They should be able to generate queries dynamically and access databases automatically whenever the need arises. In other words, service-agents should have the same data search initiation capabilities as the user and should not be dependent solely on the user for access to external information sources. In fact, the human users in such multi-agent systems may be categorized as very intelligent, multi-domain service agents. Examples of such service-agent systems can be found in the literature (Durfee 1988, Lesser 1995, Pohl et al. 1989, 1991 and 1997).

Within a networked environment the service-agents pertaining to a single multi-agent system (Fig.12) may be distributed over several computers, and even the coordination facilities (i.e., planning, negotiation, conflict detection, etc.) may be distributed over several nodes (Pohl et al. 1992). Alternatively, several single multi-agent systems can be connected. In this case each multi-agent system functions as an agent in a higher level multi-agent system. Such systems are well suited to planning functions in which resources and viewpoints from several organizational entities must be coordinated. Typical application areas include military mission planning and facilities management. The user at each node should be able to plan in multiple worlds. For example, a private world in which shared information sources may be accessed but the deliberations of the user are not shared with other users, and a shared world which allows and encourages the continuous exchange of comments, plans and instructions. The capability normally exists for the user to maintain multiple views of each world to facilitate experimentation and the exploration of alternatives (Nadendla and Davis 1995). The service-agents resident in each system (i.e., at each node) should be able to differentiate between worlds and also between the views of any particular world. This normally requires a high degree of parallelism that must be supported by the system architecture.

So far we have discussed multi-agent systems involving two types of agents; namely, service-agents and human agents (i.e., users). Other agent types are certainly feasible. Of particular interest is the agentification of the information objects that are intrinsic to the nature of each application. These are the information objects that human decision makers reason about, and that constitute the building blocks of the real world representation of the problem situation.

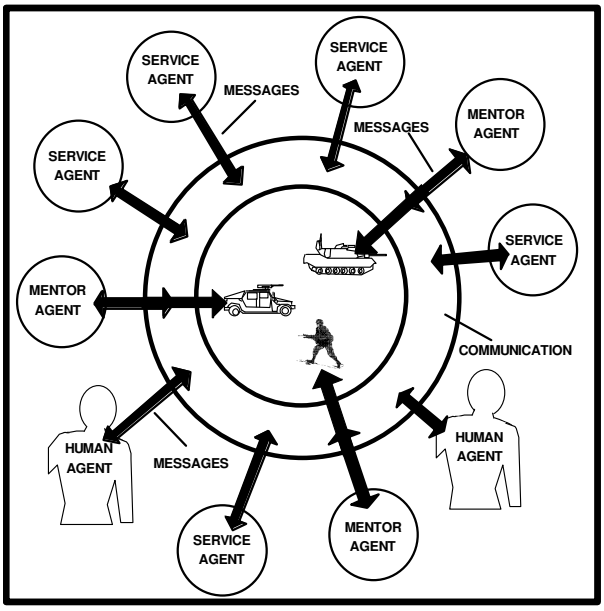

Fig.13: Object-Agent Systems

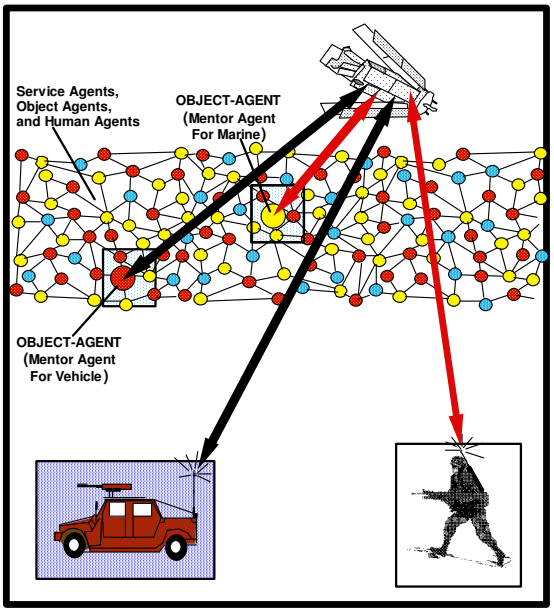

Fig.14: The Object-Agent as a Mentor

The notion of object-agents brings several potential benefits. First, it increases the granularity of the active participants in the decision making environment. As agents with communication capabilities, objects such as armored vehicles (in military missions), aircraft (in air traffic control), or building spaces (in architectural design), can pursue their own needs and perform a great deal of local problem solving without continuously impacting the communication and coordination facilities utilized by the higher level components of the decision-support system (Fig.13). Typically, an object-agent is a process (i.e., program) or component of a process that includes several adjuncts that provide the agent with communication capabilities, process management capabilities, information about its own nature, global objectives, and some focused problem solving tools.

Second, the ability of object-agents to request services through their communication facilities greatly increases the potential for concurrent activities. Multiple object-agents can request the same or different services simultaneously. If necessary, service-agents responding to multiple service requests can temporarily clone themselves so that the requests can be processed in parallel. Third, groups of object-agents can negotiate among themselves in the case of matters that do not directly affect other higher level components or as a means of developing alternatives for consideration by higher level components. Fourth, by virtue of their communication facilities object-agents are able to maintain their associations to other objects. In this respect they are the product of *decentralization* rather than *decomposition*. In other words, the concept of object-agents overcomes one of the most serious deficiencies of the rationalistic approach to problem solving; namely, the dilution and loss of relationships that occurs when a complex problem is

decomposed into sub-problems. In fact, the relationships are greatly strengthened because they become active communication channels that can be dynamically created and terminated in response to the changing state of the problem situation.

The combination of object-agents and service-agents in the same decision-support system suggests a logical transition from 2nd Wave to 3rd Wave software in which even simple learning capabilities may eventually lead to emergent knowledge (Brooks 1990). Object-Agents may represent abstract concepts such as image and power, collective notions such as climate, virtual entities such as a building space during the design process (Pohl 1996), physical objects such as a M1A1 tank in the battlefield, or even human beings such as an individual soldier, squad or platoon. In the latter case a small communication device, embedded in a computer tag, is attached to the uniform of the soldier (Fig.14). This Radio Frequency Tag (RF-Tag) is capable of receiving and sending messages to an object-agent taking the role of a mentor within the computer-based command and control system. In this scenario the object-agent can serve many functions. It can provide several kinds of assistance to the soldier, such as medical advice, geographical position and terrain information, enemy location and strength, maneuver strategies, fire support alternatives, and so on. Conversely, the object-agent can use the soldier as part of a sensory array that continuously collects intelligence with and without the soldier's direct involvement.

Many of the service requests received by the object-agent will need to be passed onto service-agents, human agents, or other object-agents. This can be accomplished through the appropriate use of both broadcasting and directed modes of communication. For example, a request for medical advice may initiate several actions by the mentor agent: a specific request for more detailed information to the soldier; the collection of bodily functions data from sensors embedded in the soldier's uniform, if the soldier has been wounded; a broadcast for evacuation assistance, if the wounds are serious; a request for specific self-help medical advice directed to a service-agent with medical expertise; a situation update to the Commander's mentor agent and/or the designated command and control service-agent; and so on. Even if the soldier is unable to personally communicate, the mentor agent is automatically alerted to the soldier's medical condition through sensors attached to his uniform or skin.

Conclusion

A collaborative agent-based command and control system, such as the ***Integrated Marine Multi-Agent Command and Control Systems (IMMACCS)*** that was successfully field-tested by the Marine Corps Warfighting Laboratory (MCWL) during the Urban Warrior Advanced Warfighting Experiment a few weeks ago (i.e., March 11 to 18, 1999, Monterey and Oakland, California), differs from the conventional *human-based* command and control system shown in Fig.10 in several significant respects (Porczak et al. 1999). First, the continuous and automatic monitoring of human/machine warfighting units by the various types of agents that operate spontaneously within the communication system potentially provides the warfighter with access to instantaneous advice and guidance. The agent to agent communication which facilitates this continuous access to information and intelligent analysis is not dependent on human to human interaction. In a conventional command and control system the communication channels are easily saturated by the continuous flow of human to human electronic and voice communications. Efforts to control this traffic inevitably require the imposition of

communication restrictions that can easily prevent critical information from reaching the appropriate Commander or warfighter. In addition, as shown in Fig.10, the human to human interaction encourages a build-up of support personnel in and around the theater. This build-up is costly in terms of transportation and logistics, increases the danger of casualties, and places an additional burden on the already overloaded communication facilities.

Second, the multi-agent system architecture decentralizes both the collection and analysis of information. Individual human/machine warfighting units serve equally well as collectors and generators of information, as they do as recipients of information. In this way a dispersed force of warfighters can represent an important sensor array, with the ability to add value by converting data into information and knowledge close to the source. This decentralization of the data analysis process is particularly valuable in terms of distributing the communication traffic and validating the results of the analysis at the collection source.

Third, the seamless integration of planning, execution and training functions within the same command and control communication system allows the Commander and the individual warfighter to continuously and instantaneously switch from one mode of operation to another. In fact, the parallel nature of the system allows specific planning, execution and training tasks to be undertaken concurrently. For example, the Commander may wish to initiate a planning function through one set of agents while executing a specific operation in the theater, and at the same time simulate a particular *what if* scenario in anticipation of a possible future situation.

Recent studies by the US Marine Corps and the US Army have demonstrated the capabilities of relatively low cost computerized RF-Tags that are mounted on vehicular cargo. Object-Agents can be designed to communicate with tagged equipment not only for purposes of monitoring their location, but also in a service and low level decision making role.

For example, let us assume a tactical cargo loadout scenario in which a fuel truck, fitted with a RF-Tag has been loaded onto a ship. During the voyage the fuel truck starts to leak. While the volume of fuel leaked is fairly small, even this small amount constitutes a serious potential hazard on-board ship. Alerted of the situation through a simple feed-back mechanism the RF-Tag communicates to its companion object-agent, resident in the command and control system, both its location and the extent of the leakage. The object-agent analyses the situation, either through its own capabilities or by requesting supporting services from other agents, and automatically notifies appropriate command personnel, or other agents, or the ship directly. ***What is particularly noteworthy in this scenario is the fact that the command and control system was not only able to automatically detect the problem, but also analyze the situation and take action without the need for human intervention.***

In existing multi-agent system configurations which include only domain agents (i.e., service-agents), conflicts arise when agents either disagree among themselves or with a decision made by the user. For example, utilizing such a system for the load planning of a ship, the placement of a fuel truck in a particular ship compartment might provoke the latter type of conflict (CADRC 1994). If the stow-planner unknowingly places the truck in the immediate vicinity of another cargo item of a different hazardous material class, then the *hazard agent* will alert the user and explain the necessary segregation requirements. The stow-planner resolves the conflict by relocating or unloading one or both of the cargo items or, alternatively, overrules the service-agent. The fuel truck, as a passive object, is involved in the conflict resolution process only as an information source that is used by the service-agent in its deliberations. In other words, while the

validation of the load planning decision is entirely dependent on the knowledge encapsulated in the object the latter is unable to actively participate in the determination of its own destiny.

There is another kind of conflict resolution scenario that becomes possible with the availability of object-agents. An object-agent may develop a solution to a sub-problem in its own domain that redirects the entire course of the overall solution plan. For example a squad, operating in dispersed mode in enemy territory and communicating with a mentor agent (Fig.14), performs its assigned enemy surveillance mission. It communicates through its object-agent certain enemy behavior that it believes could be turned to advantage if specific elements of the current overall operations plan were to be modified. However, such suggestions are rejected at operational levels below the Commander for reasons that appear to this squad to be based on erroneous intelligence. The squad judges the matter to be of a potentially serious nature and instructs its mentor agent to validate aspects of the squad's current understanding of the battlefield situation.

The object-agent commences a low level investigation by communicating with the mentor agents of several other squads and utilizing the services of domain agents (i.e., service-agents) where necessary. Soon an alarming picture emerges. It appears possible that the enemy has infiltrated one node of the command and control system and is entering erroneous information through this node. The effects of this gradually evolving deception could lead to disastrous consequences. The squad, realizing the potentially serious nature of the situation, progressively develops through the activities of its object-agent a more and more compelling case in support of its observations and suggestions.

Eventually, the overwhelming weight of evidence developed from the interactions of the squad with its object-agent and other agents in the command and control system attracts the attention of the Command Element. The Commander and his object-agent quickly undertake another analysis of the situation considering additional factors not considered in the squad's analysis. He verifies an almost certain localized penetration by the enemy of the command and control system and decides to utilize this knowledge by implementing a double-deception strategy.

This scenario demonstrates several significant capabilities of a multi-agent command and control system, like IMMACCS, incorporating object-agents. First, it is significant that the likely enemy penetration of the information system has been discovered at all. If the squad had been restricted to communicating its information as passive objects for processing by service-agents there would not have been any desire on the part of the command and control system to pursue the problem after the initial conflict resolution. Second, the squad's object-agent was able to undertake its investigation in a decentralized fashion without impacting higher level command and control activities until it was ready to present a strong case for reconsideration. However, it was able at any time to alert higher levels of the command structure as soon as the results of its investigation warranted such action.

Third, if the squad's projections had been rejected at all higher agent levels, the squad's object-agent could have appealed directly to the Commander or his object-agent. Under these circumstances the Commander would have several alternative courses of actions open: also reject the squad's suggestions; require one or more of the higher level agents (i.e., object-agents and service-agents) to explain their ruling; reset certain parameters that allow the higher level agents to reconsider their ruling; overrule the higher level agents and accept the proposal; or, capture the current state of the battlefield situation as a recoverable view and use the squad's proposition as the basis for the exploration of alternative solution paths.

Apart from their immediate action capabilities, object-agents support the highly desirable goal of decentralization through localized decision making and communication. In this kind of distributed, cooperative environment it would be useful if messages themselves could be endowed with agent capabilities. At least certain types of messages would benefit greatly from action capabilities. For example, a message-agent sent by an object-agent or service-agent to find particular information could clone itself to seek the information concurrently in several potential sources. Once apparently relevant information has been found it could be synthesized to formulate a meaningful response to the originator of the query. Clearly, message-agents would add another level of granularity, decentralization and action capability within the distributed, collaborative decision-support system architecture.

References

Archea J. (1987); 'Puzzle-Making: What Architects Do When No One Is Looking'; in Kalay (ed.) Principles of Computer-Aided Design: Computability of Design'; Wiley, New York, New York.

Bohm D. (1983); 'Wholeness and the Implicate Order'; Ark Paperbacks, London, England.

Brooks R.A. (1990); “Elephants Don't Play Chess”; in Maes P. (ed.) Designing Autonomous Agents, MIT Press, Cambridge, Massachusetts (pp.3-15).

Buchanan B. and E. Shortliffe (1984); ‘Uncertainty and Evidential Support’; in Buchanan and Shortliffe (eds.) Rule-Based Expert Systems, Addison-Wesley, Reading, Massachusetts (pp.209-232).

CADRC (1994); ‘ICODES: Proof-of-Concept System: Final Report’; Contract #: N47408-93-7347, Naval Civil Engineering Laboratory (Port Hueneme, California), CAD Research Center, Cal Poly, San Luis Obispo, California.

Duda R., P. Hart, N. Nilsson, R. Reboh, J. Slocum and G. Sutherland (1977); ‘Development of a Computer-Based Consultant for Mineral Exploration’; SRI Report, Stanford Research Institute, Menlo Park, California, October.

Durfee E. (1988); 'Coordination of Distributed Problem Solvers'; Kluwer Academic, Boston, Massachusetts.

Johnson-Laird P. (1993); ‘Human and machine Thinking’; Erlbaum, Hillsdale, New Jersey.

Kuhn T. (1977); ‘The Essential Tension: Selected Studies in Scientific Tradition and Change’; University of Chicago Press, Chicago, Illinois.

Lakoff G. and M. Johnson (1980); “Metaphors We Live By”; University of Chicago Press, Chicago, Illinois.

Lesser V. (ed.) (1995); ‘Proc. First International Conference on Multi-Agent Systems’; ICMAS-95, AAAI Press/MIT Press, Cambridge, Massachusetts.

Mackinder M. and H. Marvin (1982); 'Design Decision Making in Architectural Practice'; Building Research Establishment, Department of Environment, HMSO, London, England, July.

Mallen G. and P. Goumain (1973); 'The Analysis of Architectural Design Activity in the Working Environment'; Report 108/4 DDR, Royal College of Art, London, England, November.

Minsky M. (1982); "Why People Think Computers Can't"; AI Magazine, Vol.3(4), Fall.

Myers L., J. Pohl, J. Cotton, J. Snyder, K. Pohl, S. Chien, S. Aly and T. Rodriguez (1993); 'Object Representation and the ICADS-Kernel Design'; CAD Research Center, Technical Report (CADRU-08-93), Cal Poly, San Luis Obispo, California.

Nadendla R. and A. Davis (1995); 'FEAT: Distributed Problem Solving in a Military Mission Planning Environment'; Master Thesis, Computer Science Department, Cal Poly, San Luis Obispo, California.

Pohl J., A. Chapman, L. Chirica, R. Howell and L. Myers (1988); 'Implementation Strategies for a Prototype ICADS Working Model'; Technical Report, CADRU-02-88, CAD Research Unit, Design and Construction Institute, College of Architecture and Environmental Design, Cal Poly, San Luis Obispo, California.

Pohl J., L. Myers, A. Chapman and J. Cotton (1989); 'ICADS: Working Model Version 1'; Technical Report, CADRU-03-89, CAD Research Unit, Design and Construction Institute, College of Architecture and Environmental Design, Cal Poly, San Luis Obispo, California.

Pohl J., L. Myers, A. Chapman, J. Snyder, H. Chauvet, J. Cotton, C. Johnson and D. Johnson (1991); 'ICADS Working Model Version 2 and Future Directions'; Technical Report, CADRU-05-91, CAD Research Center, Design and Construction Institute, College of Architecture and Environmental Design, Cal Poly, San Luis Obispo, California.

Pohl J., J. La Porta, K. Pohl and J. Snyder (1992); 'AEDOT Prototype (1.1): An Implementation of the ICADS Model'; Technical Report, CADRU-07-92, CAD Research Center, Design and Construction Institute, College of Architecture and Environmental Design, Cal Poly, San Luis Obispo, California.

Pohl J. and L. Myers (1994); 'A Distributed Cooperative Model for Architectural Design'; in G. Carrara and Y. Kalay (eds.), Knowledge-based Computer-Aided Architectural Design, Elsevier, Amsterdam (pp.205-242).

Pohl J., L. Myers and A. Chapman (1994); 'Thoughts on the Evolution of Computer-Assisted Design'; Technical Report, CADRU-09-94, CAD Research Center, Design and Construction Institute, College of Architecture and Environmental Design, Cal Poly,
San Luis Obispo, California, September.

Pohl J. (1995); 'The Representation Problem in CAD Systems: Solution Approaches'; in Pohl J. (ed.) Advances in Cooperative Computer-Assisted Environmental Design Systems, focus symposium: 8th International Conference on Systems Research, Informatics and Cybernetics, Baden-Baden, Germany, August 16-20.

Pohl J. (1996); 'Agents and their Role in Computer-Based Decision Support Systems'; in Pohl J. (ed.) Advances in Cooperative Environmental Design Systems, focus symposium: International Conference on Systems Research, Informatics and Cybernetics, Baden-Baden, Germany, August 14-18 (pp.41-54).

Pohl J., A. Chapman, K. Pohl, J. Primrose and A. Wozniak (1997); 'Decision-Support Systems: Notions, Prototype, and In-Use Applications'; Technical Report, CADRU-11-97, CAD Research Center, Design and Construction Institute, College of Architecture and Environmental Design, Cal Poly, San Luis Obispo, California.

Porczak M., K. Pohl, R. Leighton, A. Davis, H. Assal and L. Vempati (1999); 'IMMACCS: Urban Warrior Advanced Warfighting Experiment After Action Report'; CAD Research Center, Cal Poly, San Luis Obispo, California, April.

Reboh R. (1981); 'Knowledge Engineering Technologies and Tools in the Prospector Environment'; SRI Technical Note 243, Stanford Research Institute, Menlo Park, California, June.

Schön D. (1983); 'The Reflective Practitioner: How Professionals Think in Action'; Basic Books.

Schön D. (1988); 'Designing: Rules, Types and Worlds'; Design Studies, 9(3), July (pp.181-190).

Senge P. (1993); 'Transforming the Practice of Management'; Human Resource Development Quarterly, Jossey-Bass, San Francisco, California.

Thornton C. (1992); 'Techniques in Computational Learning'; Chapman and Hall, Computing Series, London, England.

Wilson G., A. Cremarty, T. Adams, M. Grinberg, C. Tollander and J. Cunningham (1984); 'AI Assists Analysts in Identifying Soviet Radar Systems'; Defense Systems Review, January (pp.23-26).