

InterSymp – 2011
23rd International Conference on Systems Research,
Informatics and Cybernetics
(August 1-5, 2011, Baden-Baden, Germany)

Pre-Conference Proceedings of the Focus Symposium
on
Intelligent Information Management Systems

Tuesday, August 2, 2011

Focus Symposium Chair:

Jens Pohl
Director, KML Center, Scott AFB, Illinois
Professor Emeritus, Architecture
College of Architecture and Environmental Design
California Polytechnic State University
San Luis Obispo, California, USA

Sponsored by:

The International Institute for Advanced Studies
in Systems Research and Cybernetics
and
Society for Applied Systems Research

Professor George E. Lasker
Chairman

ISBN: 978-1-897233-77-1

Preface

The theme of this focus symposium, *Intelligent Information Management Systems*, draws attention to the increasing need for intelligent capabilities in information management software. It is somewhat of a dichotomy that on the one hand we now have enormously powerful electronic tools available and yet on the other hand these tools appear to still fall short of our needs. In fact, one might argue that in some respects they actually decrease our ability to make timely, high quality decisions.

As an explanation of these opening remarks I would like to start by paraphrasing one of my favorite authors, Charles Dickens. In *The Tale of Two Cities*, he started off the entire book with a long paragraph that began with the words: "...it was the best of times; it was the worst of times..." These are words that I believe apply very much today. We are in the best of times, because information technology and computers have become a useful partner and enabler that bring us very powerful capabilities. To mention only a few, we have: global connectivity; very fast data storage and processing devices; powerful analysis and problem solving assistance; tireless monitoring and warning facilities; and, increasingly seamless information management services. All of these capabilities greatly enable the individual. Today a single person is able to accomplish what entire organizations had difficulty accomplishing 20 to 30 years ago.

But surely, we are also experiencing the worst of times. We are driven to information system advances by very sinister forces. Suddenly, we find ourselves facing unpredictable enemies, insecurity everywhere, and revolutionary change. Our very freedom is being threatened. We are in a period of accelerated change and such periods bring about a great deal of tension. Therefore, we are also experiencing a very unsettling time in human history. What are some of these changes, and they are indeed profound changes. We are transitioning from a society that was largely governed by a sense of singularity to a society that has to increasingly deal with plurality. Most everything that we human beings have designed and produced in the past has been mechanical in nature. Mechanical systems are sequential systems. Organic systems, information systems, are pluralistic systems. They operate in parallel. So we are moving from a world that used to be paced by sequential actions to a world in which a great deal of parallelism exists.

Can there be non-human intelligence? Can the computer help us in our decision making endeavors in an intelligent partnership role? The answer to this question depends very much on our viewpoint or premises. Human beings tend to be rather self-centered. We believe that everything in our environment revolves around us. Therefore, from our human point of view, we are easily persuaded that intelligence is something that belongs innately to us. This school of thought argues that computers are electronic machines that do not and will never display truly intelligent capabilities (Figure 1). Certainly, I would agree that computers are unlikely to gain human intelligence in the near future. Several strong arguments are advanced by that school (Dreyfuss 1979 and 1997, Dreyfuss and Dreyfuss 1986, Lucas 1961, Searle 1980 and 1992). First, it is argued that humans are situated in the world by virtue of their bodies and that human level intelligence is impossible without a body. The second argument points out that symbolic reasoning and logic are not the basis of human intelligence. Human behavior is not rational and thinking does not necessarily follow rules. Third, it is argued that the world can be neither analyzed nor divided into independent logical elements. It therefore follows that the

formalization and simulation of intelligent behavior is not possible. The final summary argument of that school of thought is that for these stated reasons intelligence is the province of living creatures, specifically human beings.

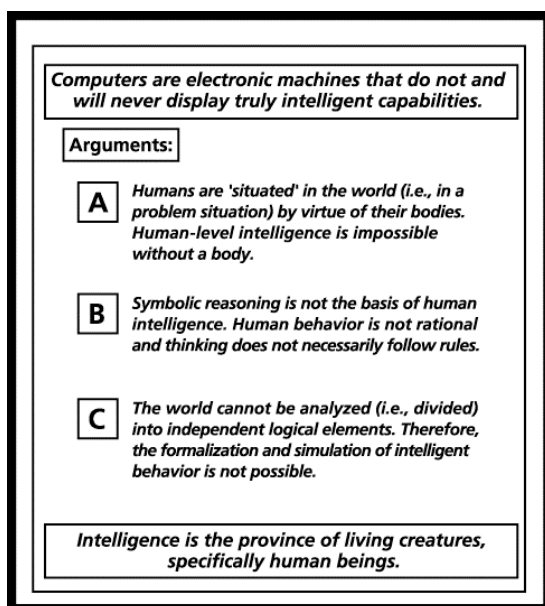


Figure 1: The human view of intelligence

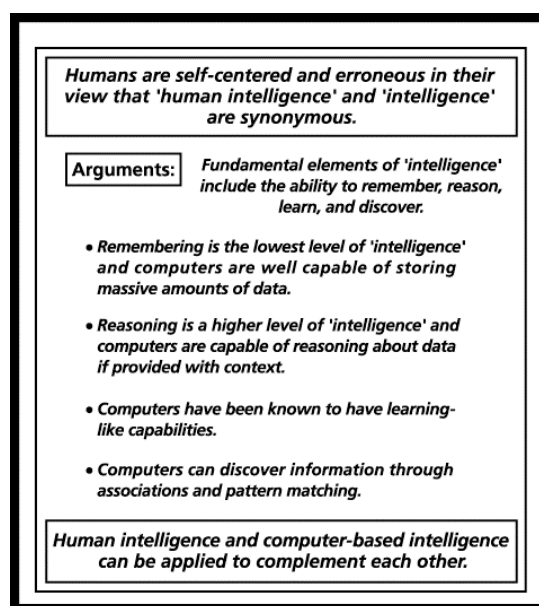


Figure 2: A general view of intelligence

A more general view of intelligence would hold that there are some fundamental elements of intelligence such as the ability to remember, to reason, to learn, and to discover or create (Figure 2). From that point of view, *remembering* as the lowest level of intelligence can certainly be accomplished by computers. In fact, one could argue that the storage capacity of computers exceeds the long term memory capacity of human beings. *Reasoning* is a higher level of intelligence and computers are capable of reasoning as long as they have some context within which to reason. However, computers cannot reason about data without *context*. Also, computers have been shown to have some learning capabilities, and computers can even discover information through association and pattern matching.

Whether there is a need for intelligent software, is the next obvious question? Until about six years ago, whenever I made a presentation on this subject there would always be a number of persons in the audience who would come to me afterwards and say: “...well this all sounds very feasible, but do we really need computer intelligence? Surely, we human beings are the ones who have intelligence and we will be able to do the necessary reasoning and interpretation of data.” Today, I rarely hear those arguments, because we are beginning to realize that we are inundated with data, and we desperately need help.

There are essentially two compelling reasons why computer software must increasingly incorporate more and more *intelligent* capabilities. The first reason relates to the current data-processing bottleneck. Advances in computer technology over the past several decades have made it possible to store vast amounts of data in electronic form. Based on past manual information handling practices and implicit acceptance of the principle that the interpretation of data into information and knowledge is the responsibility of the human operators of the computer-based data storage devices, emphasis was placed on storage efficiency rather than

processing effectiveness. Typically, data file and database management methodologies focused on the storage, retrieval and manipulation of data transactions, rather than the *context* within which the collected data would later become useful in planning, monitoring, assessment, and decision-making tasks.

The second reason is somewhat different in nature. It relates to the complexity of networked computer and communication systems, and the increased reliance of organizations on the reliability of such information technology environments as the key enabler of their effectiveness, profitability and continued existence. The economic impact on an organization that is required to manually coordinate and maintain hundreds of interfaces between data-processing systems and applications that have no *understanding* of the data that they are required to exchange is substantial. Ensuing costs are not only related to the requirement for human resources and technical maintenance, but also to the indirect consequences of an information systems environment that has hundreds of potential failure points.

Recent industry studies have highlighted the need for autonomic computing as the organizational expectations and dependence on information services leads to more and more complex networked computer solutions (Ganek and Corbi 2003). In the commercial sector “... *it is now estimated that at least one-third of an organization’s IT (Information Technology) budget is spent on preventing or recovering from crashes*” (Patterson et al. 2002). Simply stated (Figure 3), autonomic computing utilizes the *understanding* that can be represented within an information-centric software environment to allow systems to automatically: reconfigure themselves under dynamically changing conditions; discover, diagnose, and react to disruptions; maximize resource utilization to meet end-user needs and system loads; and, anticipate, detect, identify, and protect themselves from external and internal attacks.

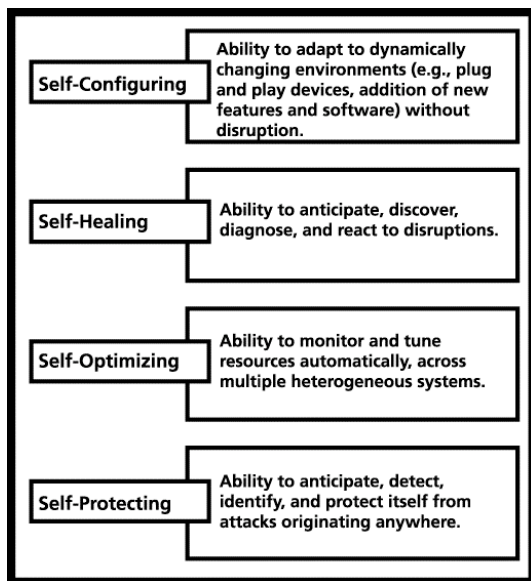


Figure 3: Desirable autonomic capabilities

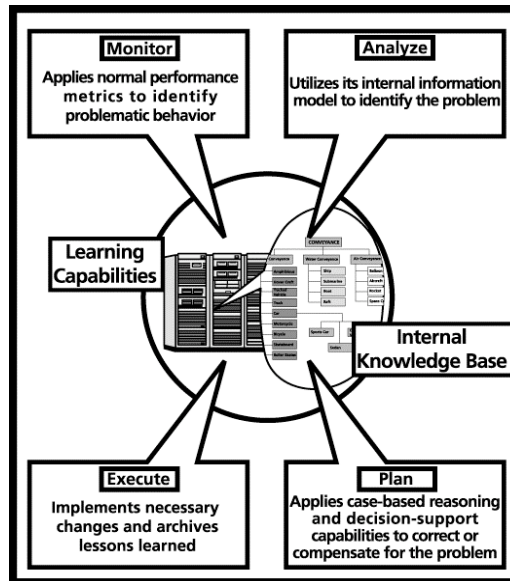


Figure 4: Autonomic self-healing facilities

These same studies have found that more than 40% of computer system disruptions and failures are due to human error. However, the root cause of these human errors was not found to be lack of training, but system complexity. When we consider that computer downtime due to security breaches and recovery actions can cost as much as (US)\$2 million per hour for banks and

brokerage firms, the need for computer-based systems that are capable of controlling themselves (i.e., have autonomic capabilities) assumes critical importance.

A core requirement of autonomic computing is the ability of a computer-based information system to recover from conditions that already have caused or will likely cause some part(s) of the system to fail. As shown in Figure 4, this kind of self-healing capability requires a system to continuously monitor itself so that it can identify, analyze and take mitigating actions, preferably before the disruption takes place. In addition, the system should be able to learn from its own experience by maintaining a knowledge base of past conditions that have caused malfunctions and the corrective measures that were taken.

Finally, the reader might wonder why the first paper in these symposium proceedings should deal with entrepreneurship, a subject that would appear to be far removed from the topic of *intelligent information management systems*. The relevance of this paper is based on the fact that difficult times such as our current deep global economic recession may either directly or indirectly force us human beings to change. They may be caused by significant technological changes or they may produce a technological revolution by forcing us to come to terms with the negative consequences of the changed environment. In either case, they provide opportunities for those of us who are willing and able to *look out of the box*. This ability to look beyond our existing *situatedness* is an essential requirement not only for economic recovery but also for necessary transition from rote data processing to intelligent information management systems.

Jens Pohl, June 2011

(jpohl@calpoly.edu) (www.cadrc.calpoly.edu) (www.cadrc.calpoly.edu/KML/)

References:

Dreyfuss H. (1979); 'What Computers Can't Do: The Limits of Artificial Intelligence'; Harper and Rowe, New York, New York.

Deyfuss H. and S. Dreyfuss (1986); 'Mind Over Machine: The Power of Human Intuitive Expertise in the Era of the Computer'; Free Press, New York, New York.

Dreyfuss H. (1997); 'What Computers Still Can't Do: A Critique of Artificial Reason'; MIT Press, Cambridge, Massachusetts.

Ganek A. and T. Corbi (2003); 'The Dawning of the Autonomic Computing Era'; IBM Systems Journal, 42(1) (pp.5-18).

Lucas J. (1961); 'Minds, Machines and Goedel'; Philosophy, 36 (pp. 120-4).

Patterson D., A. Brown, P. Broadwell, G. Candea, M. Chen, J. Cutler, P. Enriquez, A. Fox, E. Kiziman, M. Merzbacher, D. Oppenheimer, N. Sastry, W. Tetzlaff, J. Traupman and N. Treuhaft (2002); 'Recovery-Oriented Computing (ROC): Motivation, Definition, Techniques, and Case Studies'; UC Berkeley, Computer Science Technical Report (UCB/CSD-02-1175), University of California, Berkeley, California, March 15, 2002.

Searle J. (1980); 'Mind, Brains and Programs'; The Behavioral and Brain Sciences, 3 (pp. 417-24).

Searle J. (1992); 'The Rediscovery of the Mind'; MIT Press, Cambridge, Massachusetts.

InterSymp-2011
**International Conference on Systems Research,
Informatics and Cybernetics**

**Focus Symposium
on
Intelligent Information Management Systems**

Tuesday, August 2, 2011

TABLE OF CONTENTS

“The Role of Entrepreneurship, Innovation, and Intuition”	9
<i>Jens Pohl, Collaborative Agent Design Research Center (CADRC), California Polytechnic State University (Cal Poly), San Luis Obispo, California, USA.</i>	
“Technology Assessment: Applied Infocyber Scenarios”	21
<i>John Halldane, Environmental Systems Consultancy, Chino, California, USA.</i>	
“Semantic Transformation of Search Requests for Improving the Results of Web Search”	37
<i>Vladimir Fomichov and Anton Kirillov, Department of Innovations and Business in the Sphere of Informational Technologies, Faculty of Business Informatics, National Research University (Higher School of Economics), Moscow, Russia.</i>	
“An Agent Facilitated Design Conversation System for Facilitating the Designer in Creative Thinking in Architectural Design”	45
<i>Geeta Arjun and Jim Plume, Faculty of The Built Environment, University of New South Wales, Sydney, Australia.</i>	
“ICODES: A Load-Planning System that Demonstrates the Value of Ontologies in the Realm of Logistical Command and Control (C2)”	55
<i>Kym Jason Pohl, Collaborative Agent Design Research Center (CADRC), California Polytechnic State University (Cal Poly), San Luis Obispo, California, USA, and Peter Morosoff, Electronic Mapping Systems (E-MAPS) Inc., Fairfax, Virginia, USA.</i>	
“An Intelligent Supply Chain Planning and Execution Environment”	75
<i>Jens Pohl, Collaborative Agent Design Research Center (CADRC), California Polytechnic State University (Cal Poly), San Luis Obispo, California, USA.</i>	

The Role of Entrepreneurship, Innovation, and Intuition

Jens Pohl, Ph.D.

Director, Collaborative Agent Design Research Center (CADRC)
Director, KML Center (Scott AFB, Illinois)
California Polytechnic State University (Cal Poly)
San Luis Obispo, California, USA

Abstract

In recognizing the importance of entrepreneurship and innovation as the principal drivers of economic growth, this paper focuses on the human attributes that govern the behavior of the entrepreneur and the societal perceptions that influence the human environment in which the entrepreneur operates. Foremost among these human attributes is the experience-based nature of the human cognitive system that prepares us well for dealing with events that are closely related to our past experience, but forces us to learn by failure as we apply past methods to new situations. In particular, the paper discusses the difficulties that the human dependence on experience poses to the entrepreneur in terms of the innate human aversion to change, the interpretation and assessment of new situations, and the formulation of appropriate plans and strategies within the practice of entrepreneurship.

The value and pitfalls of intuition are discussed in some detail, with particular reference to the precautions that the entrepreneur should exercise so as not to be misled by the various experience-based and emotional influences that govern intuitional processes. In addition, statistical data shows that the success rate of entrepreneurial ventures in terms of actually becoming operational and the amount of personal wealth created is far below common public perception. While entrepreneurship is a leading generator of economic growth, the financial benefit to the individual entrepreneur is likely to be little more than that provided by normal employment. This suggests that the entrepreneurial urge that manifests itself in a small minority of persons, who are willing to abandon the comforts of status quo, is driven more by a combination of personality traits such as adventurism, competitiveness, non-conformance, and passion than deliberate planning based on sound market analysis. The author concludes that the critical factor of whether an entrepreneurial venture will eventually become even moderately successful depends on the willingness of the entrepreneur to learn from early mistakes and acquire the necessary knowledge and skills that appear to be a prerequisite for business success.

Keywords

Business, entrepreneur, entrepreneurship, experience, immigrants, innovation, intuition, universities.

Introduction

The business community has known for some time that any economic recession, particularly a deep global recession of the kind that we have been experiencing since 2008, is followed by an array of new products and entirely new markets that were largely unforeseen. Market analyses

have shown that entrepreneurial capabilities and opportunities are the key drivers leading to economic recovery. The willingness of individuals to *think out of the box* and take the risk to pursue an idea with passion and hard work is the dynamic backbone of a national economy. According to United States (US) government statistics more than two thirds of all new jobs were created in 2007 by businesses that were less than 10 years old (Figure 1).

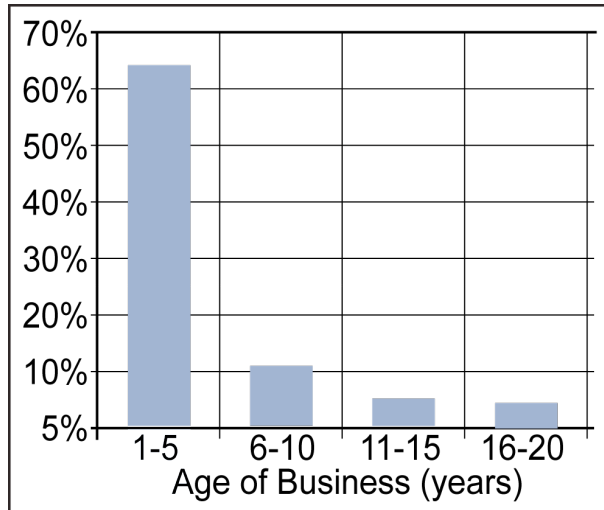


Figure 1: Job creation by business size

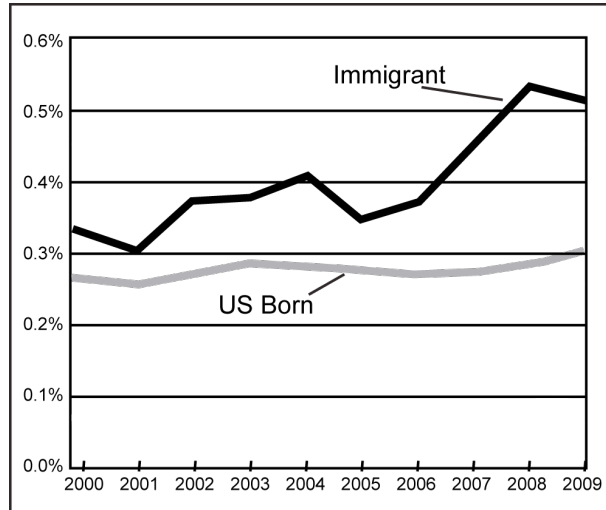


Figure 2: Entrepreneurial activity by nativity

Contrary to expectations, several economic constraints marked by lack of consumer demand and high unemployment appear to present a strong stimulus for innovation. This is consistent with past experience, which shows that the innate human aversion to change tends to be overcome more effectively when persons encounter severe difficulties in maintaining status quo. In blunter terms, the greater the pain experienced in the current situation the greater the desire to explore alternative opportunities. It should therefore not come as a surprise¹ that immigrants are disproportionately more likely to start new businesses. In the US, as shown in Figure 2, the disparity between new businesses (i.e., start-ups) formed by US born and immigrant entrepreneurs has increased significantly between 2000 and 2009 (Schramm 2011).

According to prevailing economic theory, growth of output in an economy is largely governed by the growth of input; - namely physical capital, human capital and innovation. By far the most important of these are human capital (i.e., labor and skill level) and innovation (Litan and Cook-Deegan 2011). Universities due to their educational and knowledge creation (i.e., research) roles play an important part in economic growth. Through education they add to the available skills in the labor force. More highly skilled workers are more adaptable to the dynamics of the marketplace by their generally more superior ability to teach themselves new skills. This makes them potentially more resourceful entrepreneurs. However, the need for these abilities to be applied will be most pronounced in the presence of challenges. An environment that is economically and socially comfortable is less likely to generate within individuals the strong urges for improvement that lead to entrepreneurial undertakings. This is no doubt one reason

¹ It may be hypothesized that since non-refugee immigrants have already demonstrated their willingness to take risks by leaving their country of origin to start a new life in a largely unfamiliar environment, they are less prone to adhere to status quo.

why immigrants and young persons who strive for economic respectability are key players in generating economic growth.

The relationship to universities lies in the fact that these two demographic groups tend to be more effective in generating economic growth if they are well educated. Furthermore, the higher their level of education the more instrumental they become in disseminating the knowledge that is created within universities through research. In other words, the dissemination of the knowledge that is created in universities occurs not only through academic publications and conferences, but also through the application of this knowledge when their graduates enter the workforce.

Definitions and relationships

As foreshadowed by the title of this paper the human cognitive characteristics of entrepreneurship, innovation and intuition are interrelated. Entrepreneurship is commonly defined in business terms as a pioneering activity. The word entrepreneur originates from the French word, *entreprendre* which means to undertake, such as to embark upon a new kind of business. Accordingly, the Webster Dictionary defines *entrepreneur* as a person who organizes, manages, and assumes the risks of a business or, in more general terms, an enterprise (Webster 1999, 440).

The Austrian economist Joseph Schumpeter (1883–1950) associated entrepreneurship directly with innovation leading to new manufacturing methods, products, markets, and forms of organization. In this regard entrepreneurial activities are expected to result in benefits such as the creation of new demands and wealth. In other words, the successful entrepreneur will combine various factors in an innovative manner so that the value of the result will exceed the cost of the input factors.

Intuition is one of the principal cognitive tools available to the entrepreneur to look beyond the experience of the past to what might be possible in the future. It plays a fundamental role in entrepreneurial activities because innovation is by definition a departure from existing practices and knowledge. An intuitive conclusion is not based on the deliberate and logical analysis of information that exists in our brain, but rather a leap of imagination that is typically at odds with past experience. However, the ability of the human mind to think in analogous terms by relating existing knowledge and solutions in one application domain to another unrelated domain appears to be a core component of intuition.

Humans are situated in their environment

The reason why entrepreneurship, innovation and intuition are exceptional qualities, on which we place a high value, is because they are contrary to normal human behavior. Human beings and their activities are almost entirely governed by the environment in which they exist. We are stimulated by the environment through our physical senses and respond largely in a reactive mode. These physical stimuli trigger mental processes that accumulate in long term memory as experience. Reasoning about such stimuli in the context of this experience allows us to make useful decisions as long as the environment does not change in a major manner. The nature of our cognitive processes prepares us well for dealing with events that are closely related to our past experience, but provide us with little if any means for dealing with entirely new, unforeseen events or projecting into the future.

It can be argued that we are situated in our environment not only in terms of our physical existence but also in terms of our psychological needs and understanding of ourselves. We depend on our surroundings for both our mental and physical well being and stability. Consequently, we view with a great deal of anxiety and discomfort anything that threatens to separate us from our environment or comes between us and our familiar surroundings. This extreme form of *situatedness* is a direct outcome of the evolutionary core of our existence. The notion of evolution presupposes an incremental development process within an environment that represents both the stimulation for evolution and the context within which that evolution takes place. It follows, firstly, that the stimulation must always precede the incremental evolution that invariably follows. In this respect we human beings are naturally reactive, rather than proactive. Secondly, while we voluntarily and involuntarily continuously adapt to our environment, through this evolutionary adaptation process we also influence and therefore change our environment. Thirdly, our evolution is a rather slow process. We would certainly expect this to be the case in a biological sense. The agents of evolution such as mutation, imitation, exploration, and credit assignment, must work through countless steps of trial and error and depend on a multitude of events to achieve even the smallest biological change (Pohl 1999).

In comparison to biological evolution our brain and cognitive system is capable of adapting to change at a somewhat faster rate. Whereas biological evolution proceeds over time periods measured in millennia, the evolution of our perception and understanding of the environment in which we exist tends to extend over generational time periods. However, while our cognitive evolution is of orders faster than our biological evolution it is still quite slow in comparison with the rate of change that can occur in our environment.

Human barriers to entrepreneurship

In the short term, the experience-based nature of our cognitive system creates a general resistance to change and, therefore, also to entrepreneurship (Pohl 2002). This resistance to change is exacerbated by a very strong survival instinct. Driven by the desire to survive at all costs we hang onto our past experience as insurance. In this respect much of the confidence or lack of confidence that we have in being able to meet the challenges of the future rests on our performance in having met the challenges of the past (i.e., our success in solving past problems). We cling onto the false belief that the methods we have used successfully in the past will be successful in the future, even though the conditions may have changed. As a corollary, from an emotional viewpoint we are inclined to perceive (at least subconsciously) any venture into new and unknown territory as a devaluation of our existing experience. Accordingly, the fear of failure is a severe emotional obstacle that is faced by every entrepreneur.

The absolute faith in and adherence to our experience manifests itself in several human behavioral characteristics that present themselves as potential barriers to entrepreneurship. First among these obstacles is the strong aversion to change, discussed above. Normal human tendency is to change only subject to evidence that failure to change will threaten our current existence in a significant way. Instances of the inability or unwillingness to recognize market changes driven by both technical advances and the desire of customers to take advantage of these advances abound in the business world. For example: International Business Machines (IBM) dominated the mainframe computer market but missed the emergence of minicomputers; Digital Equipment Corporation (DEC) dominated the minicomputer market but missed the rise of the Personal Computer (PC) market; Apple Corporation led the PC market with its user-friendly

computing environment but lagged five years in portable computers; and, Microsoft underestimated the importance of the Internet and had to play catch up with its Internet Explorer browser.

A second barrier is our systemic need to apply old and tried methods to new situations, even though the characteristics of the new situation may be quite unlike the situations in which the existing methods were found to be useful. This typically casts us into an involuntary experimental role, in which we learn from our initial failures. Examples abound, ranging from the development of new materials (e.g., the flawed initial introduction of plastics as a substitute for steel in traditional building structures in the 1950s) to the reluctance of the military to change their intelligence gathering and war fighting strategies long after the conclusion of the Cold War era in the 1990s.

A third barrier is our tendency to view new incremental solutions as final comprehensive solutions. A well known example of such a problem situation was the insistence of astronomers from the 2nd to the 15th Century, despite mounting evidence to the contrary, that the heavenly bodies revolve in perfect circular paths around the Earth (Taylor 1949, 108-129). This forced astronomers to progressively modify an increasingly complex geometric mathematical model of concentric circles revolving at different speeds and on different axes to reproduce the apparently erratic movement of the planets when viewed from Earth. Neither the current scientific paradigm nor the religious dogma of the church interwoven within the social environment allowed the increasingly flawed conceptual solution of Ptolemaic epicycles to be discarded. Despite the obviously extreme nature of this historical example, it is worthy of mention because it clearly demonstrates how vulnerable the rational side of the human cognitive system is to social influences (Pohl et al. 1997, 10-11).

The practice of entrepreneurship

By virtue of our experience-based biological nature we are inextricably situated in our environment and are entirely dependent on the knowledge that we have gained from interacting with this known environment. This may be characterized as a *box* within which we exist and that under normal circumstances provides us with the degree of security and comfort that we seek. Strong forces are required to overcome our innate fear of the unknown and drive us to *look out of the box*. If our current environment becomes untenable because of a serious threat to our physical safety or our social acceptance, then we may be persuaded to either attempt to modify our existing environment or find a new environment. Examples of such forces include unemployment, religious or political persecution, lack of law and order, and disease.

Apart from these negative or threatening forces there may also exist an entirely different kind of force that may precipitate change. This force is related to the inherent human desire to compete and exercise leadership that varies in strength from person to person. It is this force that drives innovation and entrepreneurship in some persons even if there are no threatening reasons why the environment should be changed. The underlying causes of entrepreneurship are therefore based more on personality traits such as opportunism, conviction, motivation, and confidence than on the fundamental human need to survive.

At face value this may suggest that innovation and entrepreneurship are human characteristics that naturally exist in certain individuals and cannot be acquired by others. This belief is promoted by the false impression that the principal ingredient of successful entrepreneurship is a

brilliant idea. In fact, as Drucker (1993, viii) points out “... entrepreneurship is neither a science nor an art” but “... a practice”. Seldom, if ever, do innovative ideas originate from random thoughts. They are normally based on the carefully monitoring of the existing environment, the identification of trends, and the agonizingly difficult task of determining the causes of these trends. Determination of the core cause of a particular problem situation is difficult because it tends to be hidden by a plethora of symptoms that were generated by the situation but are not in themselves responsible for the creation of the problem.

The practice of successful entrepreneurship is therefore dependent on a systematic process that requires the continuous monitoring and analysis of the existing environment. It cannot be too broad in scope, but must be focused on a particular subset of the environment that appears to be a cause of concern and therefore presents an opportunity for innovation. An evaluation framework will need to be created to analyze the symptoms of the problem and determine the core cause(s). This is often a tedious undertaking that requires a great deal of research, thought, and patience. Yet, it is only the very beginning of the sequence of entrepreneurial tasks that need to be performed before there can be any thought of a successful venture. However, the identification of the core problem is a critical task that will determine the eventual success or failure of the entire venture. Naturally, if the core problem has not been identified correctly then no amount of innovative thinking will lead to any worthwhile conclusion.

Once the core problem has been identified and carefully characterized the process of innovation commences in earnest. Even though *innovation* is commonly associated with some form of inspirational creativity, the word *process* is nevertheless appropriate. It involves consideration of many factors that are related not only to the core problem itself, but also to the context within which the solution will need to be implemented. Such factors include market conditions, timing, availability of expertise, cost, solution acceptance criteria, deployment alternatives, and so on. Accordingly, while the innovation process certainly requires some degree of creativity, much of the work involved is exploratory in nature. It involves careful evaluation of the factors that could conceivably impact the final solution, research into technical areas that the entrepreneur may not be familiar with, hypothesis and/or model testing, and a great deal of verbal and written communication. The documentation tasks alone can be daunting. They range from layperson explanations of the principles involved to detailed patent applications, from preliminary level of effort and budget projections to detailed cost estimates and milestone schedules, and from initial market research to elaborate business plans. While the initial concept of the innovation may have been conceived through an inspirational thought process, the translation of the inspiration into a final solution that meets most of the necessary criteria can be a demanding and time consuming undertaking. The period of time involved in the innovation process may vary from months to years and can easily derail into failure if the entrepreneur loses focus or motivation or both.

Finally, successful innovations are typically surprisingly simple. Anything new that is complex is unlikely to be successful. Indeed, the statement “... this is obvious, anyone could have thought of it” is the highest praise that the entrepreneur could wish for.

Entrepreneurship myths

There are several myths surrounding the practice of entrepreneurship. Most of these myths have been created as the result of persons trying to explain the success or failure of entrepreneurial ventures after the fact without reference to factual statistical data collected mostly by government agencies. According to Shane (2008) these myths are predominantly related to

financial issues and are promulgated as much by persons who have no entrepreneurial experience as by the entrepreneurs themselves.

Starting a business is easy! In fact, most attempts to start a company do not materialize in an operational business. According to statistical data, after seven years of operation two thirds of these companies cannot show a profit in three consecutive months.

Entrepreneurs have an intuitive feeling about where to start a business! Unfortunately, in many cases that intuition leads to failure. Many entrepreneurs do not select the most attractive industry to start a business in. There is a greater than 75% correlation between the industry selected by start-ups and the number of companies failing in that industry.

It takes wealth to create wealth! With the exception of some information technology and biotech companies, most successful start-up companies did not start with strong financial backing. Entrepreneurs typically start business ventures with little capital and very lean operations; - renting instead of buying and paying commissions instead of salaries, wherever possible.

Entrepreneurial talent rather than business type determines success! In fact, the reverse is true. The particular industry that has been selected for a new business venture is the stronger determinant of potential success and growth. While less than 0.01% of start-ups in the hotel/motel and restaurant industries have reached the Inc-500 list of fastest growing companies during the past 20 years, 4% of start-ups in the information technology industry have reached that lofty goal. In other words, the odds are at least 500 times more favorable for an information technology start-up.

Entrepreneurs become very wealthy! While it is true that entrepreneurship creates a great deal of wealth, the wealth is very unevenly distributed among only a few. According to Shane (2008) most entrepreneurs end up earning less money in their business venture than they would have been earning as employees.

Venture capital is a good source for financing a new business! Again, with the exception of information technology and biotech companies that receive about 80% of all venture capital in the US, the chances of a start-up receiving venture capital are only about 1 in 4000. Of the 3000 or so companies that receive venture capital in the US each year less than one third are start-ups.

Banks are not likely to lend money to a start-up company! According to US Federal Reserve data about 15% of all financing provided to companies that are no more than two years old comes from bank loans. Even though 15% is still a relatively low figure it is much higher than other sources such as venture capital, government grants, family loans, and other investment sources.

Clearly, the general perception of entrepreneurial enterprises by both entrepreneurs and the public is considerably at odds with reality. It would appear that our human intuition plays a significant role whenever we move from a current situation into a new situation. Even though intuition must in some manner be based on an assessment of experience, that assessment appears to be governed largely by subconscious processes. To what extent these subconscious processes are influenced by emotions and involuntary volition is not known, however, there is no doubt that the outcome can be misleading. Therefore, the next section of the paper will explore some of

the intuitive influences that can easily bias our decisions and conclusions, when there is inadequate factual information or knowledge.

Uses and abuses of intuition

Intuition is an important cognitive mechanism available to entrepreneurs as they explore innovative solution approaches. In many cases the inspiration for a particular solution will come from outside the problem area by analogy or as a spontaneous hunch that some vaguely defined idea might work. In this respect, intuition can be defined formally as the power or faculty of attaining knowledge or cognition without evident rational thought and inference. While there is still no complete understanding of the process of intuition, it appears to be a form of subconscious pattern recognition that operates largely on experience.

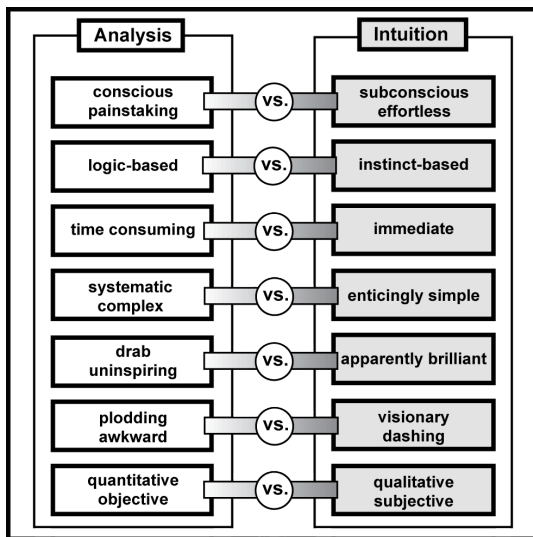


Figure 3: Intuition is very attractive

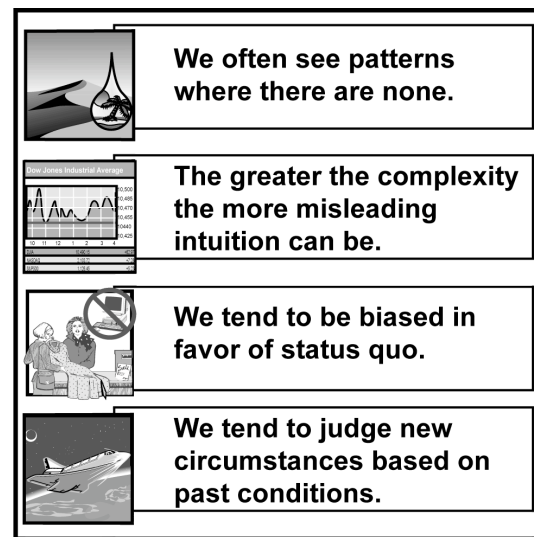


Figure 4: Intuition can be quite misleading

However, the popular perception of intuition differs markedly from this formal definition. It is typically associated with the elegance of effortless brilliance due to innate instinct, professional judgment, common sense, and superior pattern recognition (Figure 3). There are of course good reasons for this superficial perception. While analysis is consciously painstaking, logic-based and time consuming, intuition appears to be subconsciously effortless, instinct-based and immediate. Furthermore, while analysis is complex, drab and uninspiring, plodding, quantitative and objective, intuition appears to be enticingly simple, brilliant, visionary, qualitative and subjective.

In reality intuition has many pitfalls and is therefore fraught with danger (Figure 4). For example: we often see patterns where there are none; the greater the complexity the more misleading intuition can be; intuitive conclusions are often biased in favor of status quo; and, due to our experience-based nature we tend to judge new circumstances on the basis of past conditions. The entrepreneur must be aware of at least six well known decision-making dangers that are influenced by intuition.

Anchoring Trap: We tend to use the first information received as a reference point for comparing subsequent information. For example, the question “Is the distance from San Diego to Chicago greater than 5,600 miles?” will intuitively suggest to the respondent

that the distance must be somewhere in the vicinity of 5,600 miles. To safeguard against this fallacy we need to view a problem from several different perspectives and use more than one reference point. The entrepreneur should seek opinions from multiple sources and must be careful not to influence the source while asking for advice. In the above example the question would be better framed as “What is your estimate of the distance between San Diego and Chicago?”

Status Quo Trap: It is our human nature to feel more comfortable with the status quo unless there is a compelling reason for taking the apparent risk of changing. A change from existing practice or the norm will seem to be risky because the consequences of the change are not part of our existing experience. However, what appears to be a risk may not be a risk at all. The tendency is to delay or avoid the change altogether by telling ourselves to rethink this later or to wait until things settle down. This can be particularly unnerving to entrepreneurs because they are likely to be surrounded by persons who do not share their optimism of success. Entrepreneurs need to continuously reaffirm their confidence: by considering whether the status quo would be good enough if it were not the status quo; by tracing the historical path to the current status quo conditions to see how the current situation has come about; by evaluating the status quo in relationship to the expected future conditions; and, by the detailed analysis of alternative courses of action.

Confirming Evidence Trap: Entrepreneurs will be tempted to seek advice from others who have recently made decisions that are similar to the decision path that is being contemplated, even though they suspect that the advisor is likely to be biased. To avoid this pitfall entrepreneurs must be willing to carefully question all confirming evidence and be honest with themselves about their motives in seeking advice. In particular, care must be taken to avoid asking the advisor leading questions that invite confirming answers. At times this may require the entrepreneur to play devil’s advocate and force consideration of counter arguments.

Framing Trap: A poorly framed question can easily bias a decision. For example, we can be unduly influenced by risks associated with potential losses, even if there is only a remote possibility that these losses could occur. It is therefore important for the entrepreneur to consider gains and losses equally. Strategies for achieving this objective include casting the problem in several different ways and reconsidering the problem from different reference points.

Sunken Cost Trap: We are often unwilling to admit past errors in judgment and thereby can easily bias our viewpoint. Not only must we be willing to admit an earlier mistake to ourselves, but we must also allow others to admit mistakes without penalizing them. In particular, we must be willing to examine our motives and try to determine why a previous mistake may be distressing to us. In this respect, seeking the advice of persons not involved in the previous decision can be helpful.

Forecasting Trap: An entrepreneur must be careful not to be either overconfident without corroborating experience or too prudent by relying on worst case scenarios. It is necessary to take a disciplined approach in assessing the probabilities of alternative outcomes. Three strategies can be helpful in this regard. Firstly, we need to carefully examine all assumptions to ensure that none of them are biased by unusual past

experience. Secondly, it is good practice to commence the analysis by considering the extremes (i.e., the most optimistic and pessimistic outcomes). Finally, it is important to test the projected outcomes over a reasonable range of estimates.

The principal value of intuition is that it helps us to assess situations in some holistic manner based on the sum total of our past experience. The mechanism that the human cognitive system utilizes in this mental process is not fully understood. It is likely to be some form of macro pattern matching that operates at the abstract (i.e., conceptual) level rather than the logical level. We somehow develop a feeling about a certain situation that can be heavily influenced by our emotions and psyche.

Profile of the entrepreneur

Although the word *entrepreneur* is commonly associated with brilliant foresight, wealth, and effortless success, with very few exceptions quite the contrary is the case. Many entrepreneurial ventures either never reach an operational stage or are eventually abandoned for lack of financial viability. It seems that at most what the average entrepreneur can wish for is an income that is no higher than the expected salary level if the entrepreneur had continued as an employee. So, what drives entrepreneurs to forsake the comfort of status quo to embark upon an out of the ordinary and seemingly risky venture?

There appear to be at least two underlying forces that drive the entrepreneurial spirit. Firstly, entrepreneurs typically have a strong desire to be more successful than others. The competitive urge is deeply rooted in the human psyche. It has been demonstrated throughout human history in both a negative and a positive manner. As a primary cause of conflict and war it has cost the lives of millions of our fellow human beings. In sport it allows individuals to excel and serves as a source of inspiration, excitement and enjoyment to both the competitors and the spectators. Secondly, entrepreneurs are typically dissatisfied with at least some aspect of their surroundings, current situation or themselves. In this respect entrepreneurs are often restless persons who are continuously looking for something better. While this quality does not necessarily make an entrepreneur good social company, it does lead to a reexamination of existing conventions, a critical review of some piece of commonly accepted technical or scientific knowledge that may in fact be fragile, and usually results in a concerted effort to create something new.

While there may be considerable variation among individual entrepreneurs in respect to the degree to which dissatisfaction and competition drive their efforts, there is one other personality trait that is applicable to all of them; - namely a very strong work ethic. An exaggerated optimism of success forces the entrepreneur to work extremely hard to achieve this success. To be able to take advantage of opportunities that may arise in the future the entrepreneur has to prepare well beforehand. In other words, the preparations that are necessary to take advantage of an opportunity have to be well in hand before the opportunity arises. This forces the entrepreneur to undertake a great deal of work that may never yield appropriate benefits, because due to a dynamically changing environment the expected opportunity may not eventuate. The entrepreneur has to be highly motivated and extremely strong in maintaining emotional confidence, to withstand the nonchalant and disparaging comments of friends and acquaintances who of course do not see the reason for the work.

Therefore, contrary to common perception, entrepreneurs typically do not lead a comfortable life. They tend to work long hours, often not being able to fully justify the potential benefits of

their labors since the work is in preparation for future events that may never occur when viewed from a status quo vantage point. Accordingly, entrepreneurs are almost continuously immersed in what appears to be a high risk atmosphere. Since most of their fellow humans do not have the appropriate temperament or personality for this kind of lifestyle, the entrepreneur tends to lead what would appear to be a somewhat lonely life. However, driven by conviction, focused on the necessary preparatory work for the realization of future opportunities, and continuously motivated by the vision of success, most entrepreneurs would tell us that they live an exciting life of their choice.

Concluding Remarks

It appears that the majority of entrepreneurs are drawn into their ventures by psychological desires and emotional states that are based on personality traits rather than rational thought and deliberate planning. This hypothesis would provide a plausible explanation for the relatively high failure rate of small businesses and the many myths that surround the practice of entrepreneurship. Among these personality traits adventurism is likely to play as important a role as competitiveness, dissatisfaction with status quo, and unwillingness to compromise. These characteristics provide the entrepreneur with the energy to succeed but not the knowledge and skills that are required for business success. If this energy is sufficiently strong to sustain the entrepreneur from initial mistakes through a learning phase, during which the necessary skills are acquired by careful analysis and rational thought, then the chances of eventual success are greatly increased.

Statistical evidence unfortunately suggests that in the majority of cases the initial passion is either too strong to succumb to rationalization or too weak to sustain the necessary willpower for the entrepreneur to continue. It is surely surprising that even with the odds being so high against the success of the entrepreneur that the impact of those relatively few entrepreneurial efforts that do succeed should have such a significant impact on economic growth.

References

- Blanchflower D. (2004); 'Self-Employment: More May Not Be Better'; Swedish Economic Policy Review, 11(2) (pp. 15-74).
- Canter R. (1983); 'The Change Masters'; Simon & Schuster, New York, New York.
- Drucker P. (1993); 'Innovation and Entrepreneurship: Practice and Principles'; HarperBusiness, New York, New York.
- Evans D. and L. Leighton (1989); 'Some Empirical Aspects of Entrepreneurship'; American Economic Review, 79(3) (pp.519-535).
- Gilder G. 1984); 'The Spirit of Enterprise'; Simon & Schuster, New York, New York.
- Grove A. (1983); 'High-Output Management'; Random House, New York, New York.
- Kauffman (2011); 'State of Entrepreneurship (2011)'; Ewing Marion Kauffman Foundation, Kansas City, Missouri, 8 February.

- Litan R. and R. Cook-Deegan (2011); 'Universities and Economic Growth: The Importance of Academic Entrepreneurship': in Kauffman, State of Entrepreneurship, Ewing Marion Kauffman Foundation, Kansas City, Missouri.
- Mancuso J. (1978); 'How to Start, Finance and Manage Your Own Small Business'; Prentice-Hall, Englewood Cliffs, New Jersey.
- Pohl J. (2002); 'Some Thoughts on Human Nature: A Discussion of Human Strengths and Weaknesses'; Proceedings InterSymp-2002, Baden-Baden, Germany, 30 July.
- Pohl J. (1999); 'Some Notions of Complex Adaptive Systems and Their Relationship to Our World'; InterSymp-99, Focus Symposium on Advances in Collaborative Decision-Support Systems for Design, Planning and Execution, Baden-Baden, Germany, 2-7 August (pp. 9-24).
- Pohl J., A. Chapman, K. Pohl, J. Primrose and A. Wozniak (1997); 'Decision-Support Systems: Notions, Prototypes, and In-Use Applications'; Collaborative Agent Design Research Center (CADRC), Technical Report CADRU-11-97, Cal Poly, San Luis Obispo, California (pp. 10-11).
- Quadrini V. (1999); 'The Importance of Entrepreneurship for Wealth Concentration and Mobility'; Review of Income and Wealth, 45(1) (pp. 1-19).
- Schramm C. (2011); 'The Rules for Growth'; The State of Entrepreneurship Address, National Press Club, Washington DC, 8 February: in Kauffman, State of Entrepreneurship, Ewing Marion Kauffman Foundation, Kansas City, Missouri.
- Shane S. (2008); 'The Illusions of Entrepreneurship: The Costly Myths that Entrepreneurs, Investors, and Policy Makers Live By'; Yale University Press, New Haven, Connecticut.
- Shane S. (2008); 'Top Ten Myths of Entrepreneurship'; <http://blog.guykawasaki.com/2008/01/top-ten-myths-o.html>.
- Shane S. (2003); 'A General Theory of Entrepreneurship: the Individual-Opportunity Nexus'; Edward Elgar, Northampton, Massachusetts.
- Taylor S. (1949); 'Science Past and Present'; Heinemann, London, England (pp. 108-129).
- Vesper K. (1980); 'New Venture Strategy'; Prentice-Hall, Englewood Cliffs, New Jersey.
- Webster (1999); 'Random House Webster's College Dictionary'; Random House, 2nd Edition (April), New York, New York.

Technology Assessment: Applied Infocyber Scenarios

Dr John F. Halldane

Principal : Environmental Systems Consultancy

e-mail: jfhalldane@hotmail.com Tel: ++1-909-949-2168 CA USA

Overview :

CADRC and KML present an urgent need for analyzing the enormous volume of digital data in people centered applications. Adaptive knowledge-based collaborative agents were suggested. A matrix for collaborative control, monitoring and management was outlined (Halldane Aug 2006) based on inline and crossline management principles. This introduced *channels* describing the technology and context, *nodes* of common parameters or attributes to link with other channels through secure analytical *gates*, then parallel *tracks* of meaningful criteria from performance, specifications, monitoring, to priorities. This is now applied in analytical *scenarios* to compare channels and tracks for assessing technology against meaningful infocyber tracks (Figure 1).

Technology assessment, TA, sets science-based overlays of scenarios for each meaningful parameter to be compared. Conclusions are drawn from each scenario according to the context and priority for the linked parameters in the technology assessment. Meaningful TA has been a tradition in management, especially for the US astronaut moon landing in 1969 spurred by the sputnik dogs in 1959, the creation of an “impact statement” culture in the 1970’s and the energy scenarios for the Project Independence Evaluation System, PIES, presented to Congress in 1974. Unfortunately TA has degraded from the 1990’s with fake pseudoscientific public-media-political agendas, particularly by environmentalists and green movements. Those assessments ignore the analysis of basic economic considerations, lifecycle costing, maintenance, performance efficiency and tangible impacts. There are further confusing features in their future agendas (US Green Building Council, USGBC) with a more focused approach to “social equity” and an increasing activity in government subsidies, tax credits, control, regulations, litigious solutions and conflicting design criteria.

Thus this paper outlines the development of meaningful scenarios, methods of assessment and tangible priorities for today’s technology assessment based on viable science and responsible management. An example of an inefficient, costly, poor investment solar photovoltaic system for a classroom is used to illustrate the principles and to highlight the issues with alternative solutions.

Working procedure in Technology Assessment :

Scenario development needs a consistent working procedure in order to manage the infocyber. Refer to the summary diagram for scenario overlays in Figure 1.

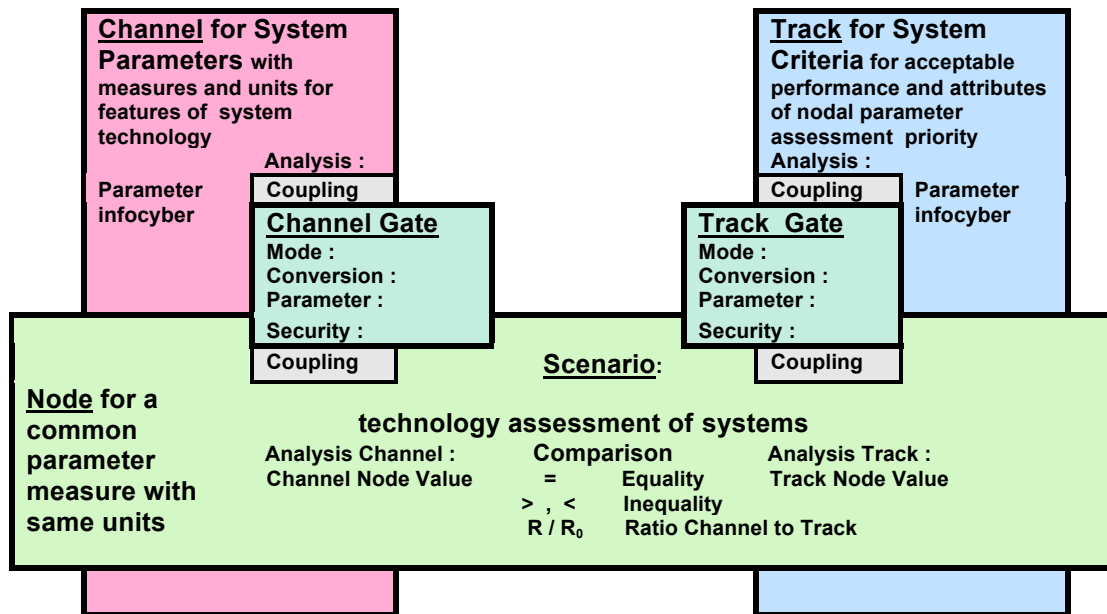
- a. Forming scenarios :** Identify the technology, context and objectives for assessment.
Channels: Define and model the parameters of systems for the technology and context.
Tracks: Determine meaningful and acceptable parametric criteria to assess the systems
Gates: Determine the access, bias and security for infocyber, marketing, research, testing.

Technology Assessment of Infocyber Systems

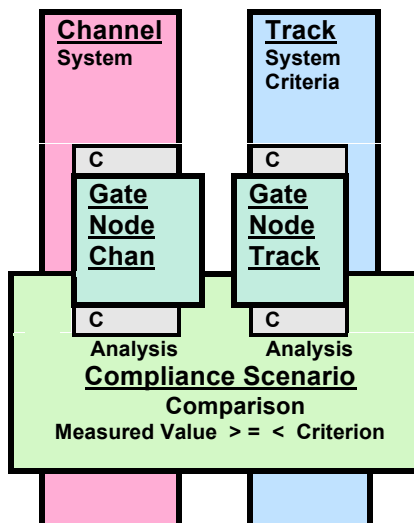
Dr John F. Halldane 11.2010

Figure 1 : Channels . Nodes . Gates . Tracks . Scenarios

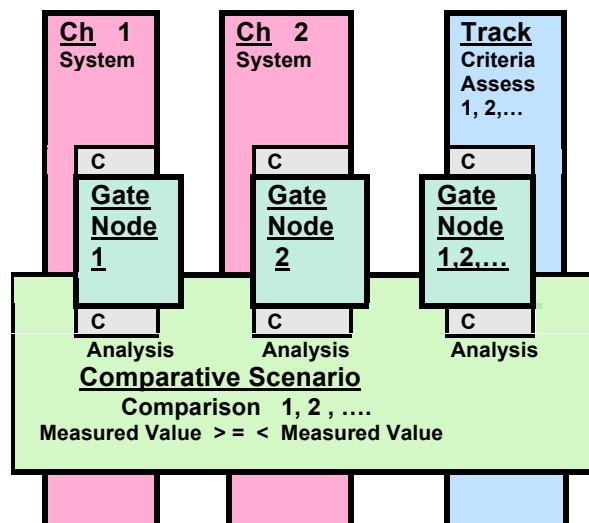
- Technology :** Description of a working system in terms of the parameters, measures, attributes,....
- Assessment :** Comparison of system with performance criteria or with other systems for similar performance.
- Infocyber :** Information and cybernetics related to the systems involved in the technology assessment.
- Channel :** Organization of the system describing the specific technology and the context.
- Node :** Common parameter or side by side measure, linking the channels and tracks for comparison.
- Gate :** Coupling mode, access, control and security of infocyber between channels, nodes and tracks.
- Scenario :** Comparable scene for system parameter to be assessed through the node. efficiency, cost,..
- Track :** Meaningful performance criteria or values for the assessed parameter...
Selection priority,... specifications, standards, security, comparative significance,...
Monitoring and maintaining system performance from a parallel independent channel.



Compliance Scenario



Comparative Scenario



Nodes: Develop scenarios from the common parameters to compare channels and tracks.

- b. Defining parameters :** Determine meaningful attributes, measures, units and relating

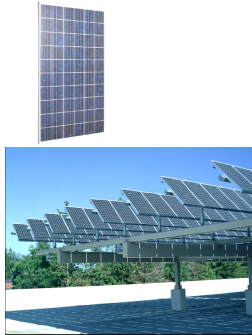
models relevant to each nodal scenario overlay, such as power P , efficiency k , energy W , costs C , resources used, impacts, contaminants, for the whole identified technology. Use subscript notations to qualify measures.

- c. **Measuring parameters :** Determine the values for the parameters in the context of the working system. Technical specifications from the manufacturer. Performance testing of system in operation. Monitoring infocyber from operating systems. Maintenance logs. Experimenting by measuring in field under varying conditions. Simulating systems and mockups. Performance, consumer and market surveys. Instrument accuracy best within 5% but this can be difficult with varying field conditions. Basis for deductive logic models.
- d. **Illustrating relationships :** Diagrams from venn, flow, to math functions. Multimedia presentations of analogous working systems, simulations. Prototypes of system components. Inductive self-evident logic.
- e. **Formulating relationships :** Each nodal scenario has a gate to analyze infocyber relationships from channel to node as a source of comparisons for that scenario. In engineering and design the relationships are formulated and published in handbooks, standards, codes to professional practice. The issue is to apply them to the systems in an analogous meaningful context. These standards and codes should be revised as technology and criteria evolve. In marketing, maintenance and facility management relationships may be tenuous so often market and consumer surveys are structured with appropriate statistical interpretation. Scatter diagrams of statistical and variable data are graphed and analyzed for regression functions with error. Here a simple “middle third” method with 87% confidence about 10% error is sufficient in systems design and assessment.
- f. **Analyzing scenarios :** At each gate relevant infocyber with their measurement in the correct context are applied to the formulated relationships according to their units of measure.. The resulting measures at each channel and track node are compared for “greater or less than or equal to” criteria in a side by side inequality. A ratio against the track or alternative solution channel can quantify this disparity. With cumulative or integrated data there is often a threshold potential, temperature, voltage or control for the system to work. This is often overlooked in natural resource utilization such as solar, wind, rain.
- g. **Assessing scenarios :** A matrix of scenario overlays are formed that connect through the nodal gate analysis. For instance in separate scenarios, the power ($\text{Watt} = \text{Joule/sec}$) of a system determines its size which in turn determines the capital cost (\$). However, the energy consumed or work done (Joule) for a system in time (year, Joule/year) determines its use or consumption which in turn governs an annual cost ($\text{\$/Year}$). These 4 scenarios again create a further lifecycle costing scenario ($\text{\$/lifecycle}$) overlay along with additional maintenance and operating costs. Although each scenario assessment (power, capital, energy, annual cost, lifecycle) is independent they can be dependent through the nodal gate analysis. Priority for which scenario is important depends on the bias in vested interest of the parties to the assessment. Investors and owners want a fast payback with residual value or salvage at the end of a lifecycle. Manufacturers and contractors need quick sales without maintenance issues. Customers want hassle free, economical, well performing, low maintenance systems. We discuss the ethics of priorities with extraneous issues at the end of the paper. When systems fail to meet criteria better alternative solutions should be suggested. Analysis is disseminating scenarios. Synthesis is integrating scenarios in design.

Technology Assessment Procedure : An Illustration : Solar Photovoltaic System for Classroom Lighting :

- a. **Forming scenarios : Description and context** of system to be assessed : The components for a solar photovoltaic system for classroom lighting is illustrated below. Solar power-energy is converted with photo cells to low voltage DC or direct current electricity, stored in batteries, then inverted to a higher voltage 120V AC or alternating current to offset the power from a utility electricity grid to the luminaires. There are significant technical and operating issues along the way.

Solar panel array :
Photovoltaics, PV :



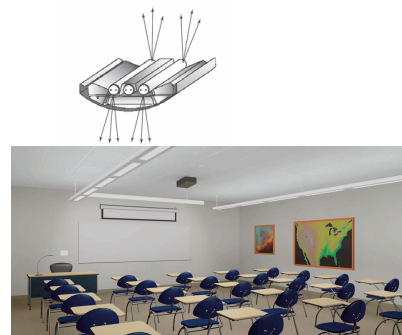
Battery bank :
Controller :



Inverter :
Grid connect:



Luminaires :
Classroom illumination :



Objectives : 1. To assess performance, operation, economics and impacts of the described system in offsetting the utility power supply. 2. To suggest alternative comparative systems.

Channels : Power distribution efficiency from potential solar to lighting. Energy and storage distribution efficiency from resource to offset grid electricity. Capital costs from power needs and construction. Annual costs from offset energy, operations, maintenance. Lifecycle costs with financing for payback on investment. Contaminants from manufacture to operations. Alternative systems for comparisons.

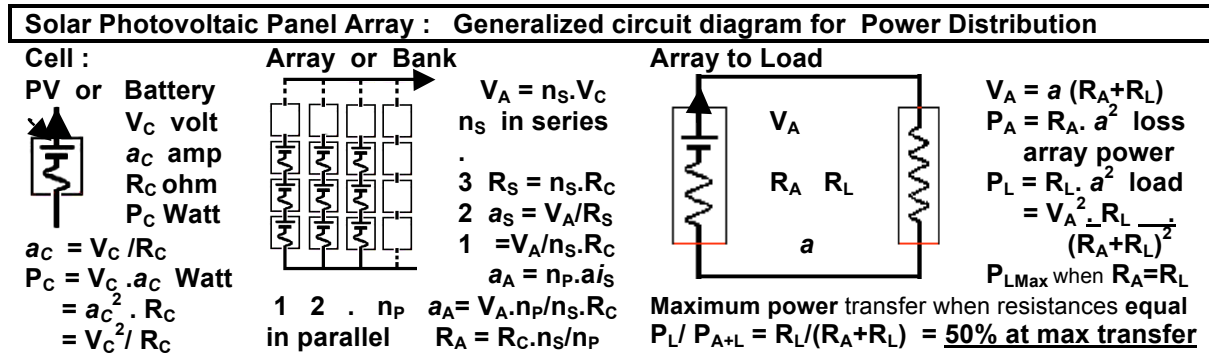
Tracks : Acceptable criteria for comparing with channels. Product specifications. System standards and codes. Case studies.

Gates : Infocyper filtered through biased sources according to vested interests from selling whole systems to separate components. As a new applied technology with few monitored demonstrations it is difficult to find consistent infocyper. Our TA approach is through using an integrated self-evident empirical scientific logic.

Nodes : We focus on compliance scenarios comparing a generalized system with the available general track information. Alternative solutions just compare those separate channels.

- b. **Defining parameters :** voltage V volt, current a amp, resistance R ohm, power $P = V.i$ Watt (Joule/sec), time T hour, energy or work done $W = P.T$ kWh (Joule) = $V.a.T$ amp-hour a-h, efficiency $K = P_{Out}/P_{In} \%$, $K = W_{Out}/W_{In} \%$, length ft ($\approx 0.305m$), area A ft² ($\approx 0.093m^2$), unit density W/ft^2 $\$/ft^2$, $E = P/A$ W/m^2 , capital cost C_C \$, annual cost C_Y \$/year, lifecycle cost C_L \$, payback period T_{Pay} year, resources used, impacts, contaminant concentration.

- c. **Measuring parameters** : In this assessment field measures are not undertaken. Typical performance and design specifications are used from available inforcyber.
- d. **Illustrating relationships** : combined with
- e. **Formulating Relationships** : Forming track channel criteria.
Models : “Design integration for minimal energy and cost” Halldane, Elsevier Pub



Cells must never have a reverse current, shorted nor fully discharged. Batteries should be fully charged and never below 75%. Note series cells n_s build voltage, parallel cells n_p reduce current which then reduces cell heating and improves performance.

Maximum power transfer is when resistances the same for array to inverter or to charge a battery bank. Also with bank discharge to inverter the loss heats the batteries.

Solar Irradiation : Photovoltaics <i>need blue clearsky sunshine</i> without cloud to work						
Power Irradiance normal to array — $k/\sin\Theta_{\text{AltitudeSun}}$ $E_{AN} = 1350 \text{ W/m}^2 \bullet e$ Atmosphere absorption $k = 0.24$ Summer, 0.12 Winter $\square_{\text{AltitudeSun}} = \text{Altitude of sun}$ $\square_{\text{Lat}} = \text{Latitude}$ Maximum power density normal to array panels may be considered as $E_{ANSunMax} \approx \underline{1000 \pm 25} \text{ W/m}^2$ peak Summer $\underline{1000 \pm 80} \text{ W/m}^2$ peak Winter Over 40° Latitude, NY, Beijing, Tasmania, Rome, Pampas, Wellington NZ, ... it falls off rapidly. A clear winter atmosphere offsets a lower sun angle. Daylight hours are shown.	$\Theta_{\text{LatSum}} \text{ } k=0.24\text{Sum}$ $E_{ANMSum} \text{ } T_{\text{DaySum}}$		$\Theta_{\text{LatWin}} \text{ } k=0.12 \text{ Win}$ $E_{ANMWin} \text{ } T_{\text{DayWin}}$		Peak \square_{AltM}	
	0	1061	12.1			90
	10	1061	12.7			90
	20 ^o	1061	13.5	20 ^o	1200	90 ^o
	30	1058	14.4	10	1195	80
	40	1046	15.1	0	1188	70
	50	1023	16.3	10	1175	60
	60	987	19.4	20	1154	50
	70	929	22.0	30	1120	40
	80	835	24.0	40	1062	30
	90	669	-	50	951	20
				60	407	10

Value $E_{ANSunMax}$ is for a tracking panel array normal to sun. For a fixed array sun is not normal
 $E_{ASunMax} = E_{ANSunMax} \cdot \cos \phi_{NSun}$ angle normal to sun. $\phi = (T - T_{\text{Noon}}) 90 / T_{\text{Day}}$

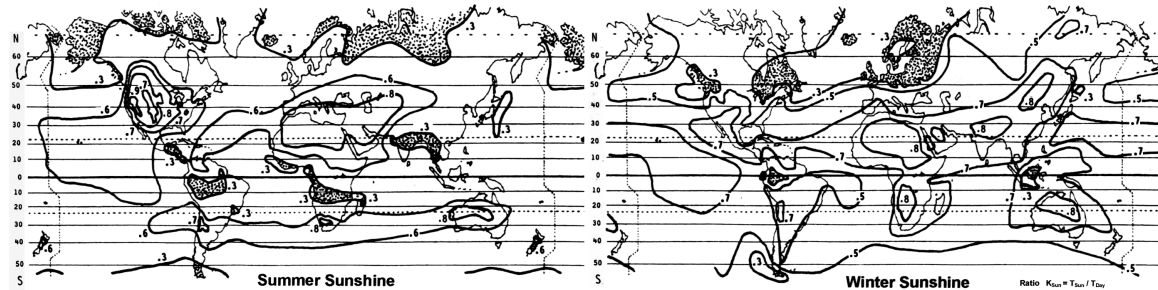
Irradiated peak solar Power $P = E \cdot A_A$ = power x area of array

$$P_{SunM} = E_{ASunMax} \cdot A_A \approx 1000 A_A \cdot \cos \phi_{NSun} \quad \text{Watt} \quad (\text{W/m}^2)\text{m}^2$$

For a more detailed account look up table for your latitude for summer and winter E_{ANMSum} , E_{ANMWin} to gain 10% in winter. The flip in winter values is for equatorial latitudes in a 20°

suncone. The corresponding peak sun altitudes are in the right column which is also used in sunshading. This power calculation is for sizing the system and consequently the capital cost.

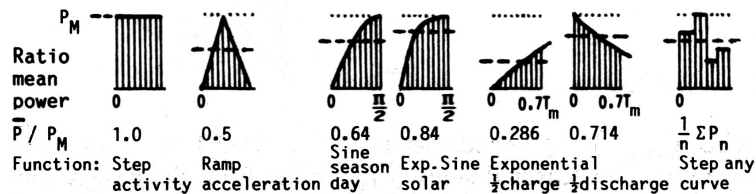
Available Solar Irradiance : Summer and Winter Sunshine 200 W/m² burn threshold



Sunshine Ratio $K_{Sun} = T_{Sun} / T_{Day}$ compares time the sun is out with the total time of day...

$K_{Sun} = 0.3$ worst in rainy humid tropical cloudy summers : Panama, Amazon, Congo, India...shaded
 = **0.4-0.5** rainy overcast : UK, Ottawa, South America, Roaring Forties, Russia
 = **0.6-0.7** most temperate climates : NZ, Australia, Florida, Europe, LA, China
 = **0.8** good in clear winter deserts : Kalahari, Sahara, Himalayas, Gobi, Kimberley
 = **0.9** best in high mountain summers : Sierra Nevada,

Estimating Mean Profiles as a portion of their enclosing rectangle.



Irradiated solar Energy (Work) $W = P \bullet T$ = power x time kWatt-hour kWh , kJoule = (kJ/sec)sec

Thus energy needs to **integrate the peak power** in terms of **array, day, season and available clear sky**. Integrating a tracking array $\cos \phi = 1$ but a fixed array profile has a $k_\phi = 0.64$ **mean**. As the daily sun altitude angle lowers from a peak, the sun power lowers by the exponential sine so the effective daily profile is $k_{Alt} = 0.84$ **mean**. The seasonal peak power $E_{ANSunMax}$ moves between summer and winter so the seasonal mean is for the Latitude $\Theta_{LatSum} +10^\circ$ in summer and Latitude $\Theta_{LatSum} -10^\circ$ in winter. Likewise for the day time hours T_{Day} the daily mean is for Latitude $\Theta_{LatSum} +10^\circ$ in summer and Latitude $\Theta_{LatSum} -10^\circ$ in winter. The available sunshine ratio K_{Sun} is determined from the map or local data. For energizing lamps, charging storage, offsetting grid.

As an example for calculations : consider a **100 m² fixed array (1076 ft², 32'x32')** in New Orleans **30°N** Note hurricane season Aug-Sep cutting into both solar seasons.

Peak Solar Power : $P_{SunM} = E_{ASunMax} \bullet A_A$ 1058x100 = **106 kW sum** 1120x100 = **112 kW win**

Solar Irradiation Energy :

Summer $W_{Sum} = 104.6 \text{ kW} \times 0.64 \times 0.84 \times 15.1 \text{ h} \times 0.65 \times 183 = 101,003 \text{ kWh sum}$

$\Theta_{LatSum} +10^\circ$ k_ϕ k_{Alt} T_{DaySum} K_{SunSum} Days

Winter $W_{Win} = 115.4 \text{ kW} \times 0.64 \times 0.84 \times 10.9 \text{ h} \times 0.50 \times 182 = 61,537 \text{ kWh win}$

$\Theta_{LatWin} -10^\circ$ k_ϕ k_{Alt} T_{DayWin} K_{SunWin} Days **162,540 kWh/y**

Power Efficiency of photovoltaic cells K_{Cell} is about **20%** for most **newer silicon** multicrystalline based cells. It is a current system with about a constant voltage output. A cell **100 cm²** will produce about **1.5 W** of power at **0.5 V DC** at **3 a** amp under **1000W/m²** sun with resistance **0.17 ohm/cell**. Active cell area is about **0.007-0.01 m²/W**. Manufacturer specifications are quite varied depending on the panel voltage and current rating then array configuration, for example, with **1000 W/m²** irradiation assumed

Panel	215 W, 29 V	at max power,	7.4 a	at max power,	400 \$, 1.8 \$/W
	195 W, 17.6 V		11 a		330 \$, 1.6 \$/W
	1.2 m² area				21.7 V open circuit

Panel Performance degradation can be significant at **2.5%/y** with panel yellowing, cracking glass, corrosion in wiring, dry joints. If cells go out or there is uneven shading with snow/leaves, a back current can develop which destroys those cells in series. Warranty periods, often 5 years, are only for parts so performance may only be **50% in 20** years and there is no residual asset for resale. Another disturbing feature is that companies may not last to service and maintain the products.

Energy Storage Batteries convert electrical energy into chemical when charging and the reverse on discharge. To charge a battery takes **60-80% charging power** to chemically deposit lead sulfate PbSO_4 and PbO_2 on lead electrodes with water to store. The reverse reaction takes **75-85% discharge power**. Together the efficiency is $70 \times 80 = 56 \pm 8 \%$ Controls limit the state of charge SOC to **50- 60-100%** to avoid overcharging and fully discharging which can damage the battery from overheating, gassing, and sulfation. Storage is rated in amp-hour, *a-h* then times the voltage gives energy in Watt-hour, W-h

A **12V** battery with **200 a-h** has a capacity of **2400 W-h**.
 If a battery takes **20 h** to drain completely with an **8 a** load the amp-hour is $8 \times 20 = 160 \text{ a-h}$.
 Operation time for a **12V, 160 a-h** capacity, with **15 a** load will last $160 \text{ a-h} / 15 \text{ a} = 10.67$ hour.

A typical specification : **12V** nominal, **12.9V** full charge float, **11.4V** fully discharged float **12.6-13.8V** at *a-h/5* charge voltage, **12-10.2V** at *a-h/20* discharge voltage

Dimensions : **12V, 126 a-h, 13 x 6.9 x 8.5"** (33 x 17.5 x 21.6 cm), Warranty 1 year, Weight 74 lb 170\$

A battery with **60%** of its capacity left is considered worn out, life **500-800 cycles**, warranty **3-5 y** for parts but not performance.

Choosing Battery Capacity involves more than multiplying the load current by the backup time in hours. First de-rate the battery for capacity tolerance, temperature, and discharge rate.

- Multiply the average load current by the backup hours of operation needed.
- Add 15% to cover loss of capacity from tolerance and UN-cycled batteries.
- For every 10°C (18°F) below room temperature (72°F) your worst case low temperature is add 10%

- If your back-up time is less than 20 hours, add 10% for every time you have to double your back-up time to equal more than 20 hours. For example : 20 minutes would be doubled 6 times to equal more than 20 hours. Add 60% on to the required capacity.
- Add 40% degradation for an economic life cycle. 60% of its capacity left is considered worn out.

Example : **10 Hours at 200 ma**, average current, worst case temperature is 0°C

Backup time ...10 hour x 0.2 a current...	2.0 a-h	
Tolerance loss... 15% x 2.0 a-h ...	0.3	
Temperature ...0°C 20% x 2.0 ...	0.4	
Backup<20 h ... 10% x 2.0 ...	0.2	
Degradation... 40% x 2.0 ...	0.8	
	Total	3.7 a-h 185 %

Inverters convert array and battery DC to load AC and voltage. The voltage step is often nominally **24V** array and bank to **120V 60 Hz** load depending on the series-parallel circuit currents with 94% efficiency x90% power factor = **85%**. Dimensions 5kW 28.5 x 15.9 x 5.7” (75.5 x 40.3 x 14.6 cm)

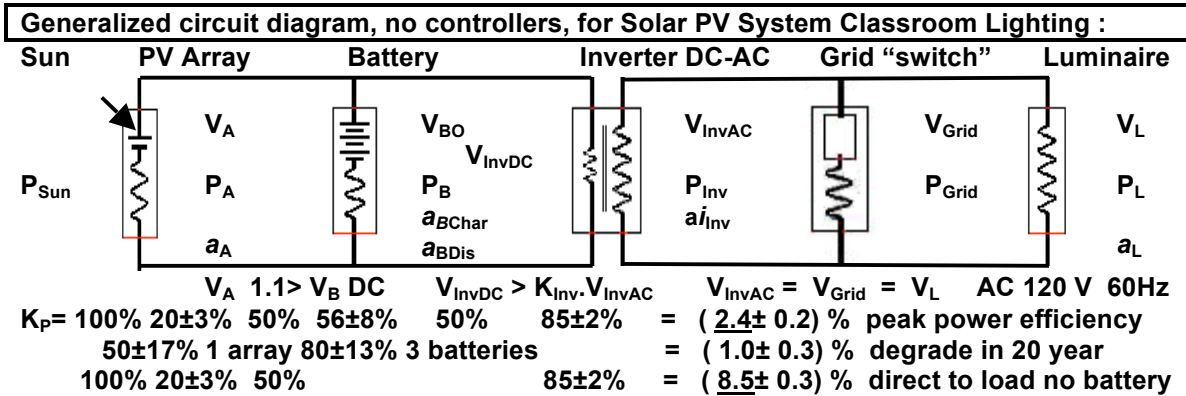
Grid-“switch” is simple on-off connection, like a light switch, to use when the array and battery has insufficient power to drive the lighting Best with steady loads rather than intermittent use.

Grid-tie is essentially a “watch the meter run backwards” and “sell back to the utility” concept. There are issues from connection fees, control installation, maintenance and legal responsibility. The advantage is that the array can always dissipate energy when the lights are off and when batteries are fully charged. Never use utility energy to charge the batteries as the losses are huge, but more importantly storage increases pollution at the power station. Utility efficiency, plant-grid-load, is about... fuel 100%, steam 60%, turbine 80%, generator 90%, transmission 95% = **41%** and if stored... inverter 94%x90%, battery 70%x80%, inverter 94%, circuit 50% = **9%** that is about **5 times the pollution** at the plant if powered through battery storage. This also defeats the argument for electric cars as they may save pollution on the road but add over **5 times** as much at the power station.

No-grid stand alone uses array and battery to load, usually for remote applications.

Lighting Load for classrooms by energy codes must be less than **1.1 W/ft²** connected to the utility. With electrical ballast loss about **10-15%**, luminaire features, room distribution and task illumination performance about **30-50 footcandle fc**, design specifications can become quite tight. So the lampwatt power density should be **0.8-0.95 W/ft². 8.6-10.2 W/m²**. Generally a **120VAC 32W T8 4ft lamp** is used in a 3 lamp luminaire in two rows on 14ft centers with instant start electronic ballasts, ballast factor 0.88. Our scenarios do not consider fixture costs nor ballast loss.

Integrating the circuits



Efficiency K is a ratio of the power out P_{out} to power in P_{in} of a system $k = P_{out} / P_{in}$. The output of a series system, with one going into the next, is the product of all those systems $k = k_1 k_2 \dots$. Efficiency determines the **extra system size and capital** to compensate for the losses in an inefficient design such as in poor fuels, storage, low power devices, poor maintenance,....

$E_{DemT} \text{ (J)}$ Energy Demand	=	$\frac{P_{Dem} \text{ (W)}}{\text{Power Demand}} \times \frac{T \text{ (s)}}{\text{Time Use}}$	=	$\frac{E_{Den} \text{ (Joule)}}{\text{Energy Density Supply}} \times \frac{Q \text{ (V,A,m)}}{\text{Quantity Resource}} \times \frac{k \text{ (W/W)}}{\text{Efficiency}}$
Energy needed for the time in use ; month for utility billing, annual for economics.				Energy per volume, area, mass within resource Amount resource used System distribution efficiencies

Capital Cost, C_{Co} is the original cost to plan, design, select, finance, manufacture, transport, install, run, test and commission a system. System capital ranges from an off-the-shelf product right up to a custom made design. **Capital is based on the power** that is needed to drive the system. The most often overlooked capital is the **financing**, particularly with new technology, in retrofitting before the end of a useful life and in underestimating a construction or maintenance budget. Energy performance contracts suffer here as they need to guarantee savings to repay their investors over time. As a consultant between parties I have found these contracts are rarely successful because of discrepancies in monitoring the savings, changes in use-occupancy, poor design and equipment, lack of maintenance, companies folding, .. essentially the building owners end up paying the bills. Capital costs are only seen by the owners and shareholders in the precommissioning phases, then in renovation or refitting to upgrade technology during occupancy. There are also strategies to offset power related capital to portions of approved building construction costs for refitting and renovation. **Government incentives** are best with a **manufacturer as investment credit** to pass on as a price rebate. Subsidies provide no payback for the taxpayer as an investor. Income tax credits and deductions are only proportionally good as the taxpayer income tax bracket.

$C_o \text{ (\$)}$ Cost Original	=	$P_D \text{ (W,A)} \cdot \frac{C_P \text{ (\$)}}{\text{Power Demand Unit (W,A) Cost}}$	$[1 + R_{Com}(\%) - R_{Dis}(\%) - R_{Gov}(\%)]$ Rates for Sales Commissions Discounts Incentives	$[1 + R_{Aux}(\%)]$ Auxilliary Systems
Original cost of product.		System size power, area Cost rate equipment power	Wholesale plus: Commission, markup, royalty, ... less : Discount, rebates, tax credits Government manufacture incentives,	Portion extra equipment, housing

Annual Cost C_Y is the sum of the yearly costs related to the energy used in the system through time and the **repayment of the financing costs**. Repayments are often neglected because they involve the owners rather than the facility managers who are responsible for the cash flows in daily operations. Managers tend to think of savings as profit for a business. A business **apportions a budget for a monthly utility bill to run a local conventional system** needed for that occupancy. When **utility bills exceed about 8% of their revenue** then managers tend to seek conservation methods to reduce those utilities but still within their budget. It is the owners of the facility who may seek capital intensive conservation to reduce those bills, however in rented-leased facilities there is little incentive to finance an upgrading by owners. Utility cost depend on the price of their resources, peak load periods, seasonal demand, subsidies for low income families.

Lifecycle Cost C_L is a balance in the **sum of the annual costs with capital in time** as compared with a conventional economic solution for that same business. A **payback** period is often used to determine the **rate of return on an investment** from an investor's viewpoint. Economists use discounted costs or present worth which ask what do I invest now to make a certain amount in the future. **Discounting should never be used in budgeting** because it underestimates real costs. It possibly explains why some government agencies are short changed in their budget requests since the Office of Management and Budget, allocates resources based on discounted economics. We **compare lifecycle scenarios for periods beyond the useful life of the systems** to include the cost of replacement or refitting C_{Ref} up to the best systems lifecycles. In estimating future costs a monetary inflation rate R_{Mon} is used and sometimes deflation R_{Def} where system prices reduce with market competition.

Payback Period T_{pay} for conserving applications is the time when the **sum of the annual cost savings balance with the extra capital** needed to create those savings **compared with an existing or conventional case**. An investor payback period is simply the time they get their money back with interest, dividends, or consideration. Most lenders want a 3-5 year payback

with 7-12%/y interest and longer lower interest venture capital is extremely difficult without collateral security.

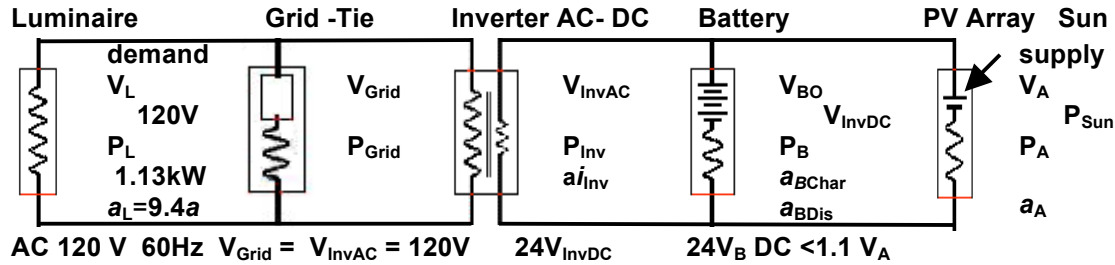
Conservation strategies come from changing the values of the parameters in the lifecycle model so that they balance in time. The simplest is to **reduce the power-time of the demand** without buying extra equipment; turning off lights and air conditioning, open windows,... A critical parameter is in **sustaining the efficiency k** of systems with degradation and aging; one guide is the IRS depreciation time which is about 12-15 years for electrical and mechanical equipment 1 year for computers, fluorescent lamps 80% from initial lumens, polycarbonate transmission loss by weathering 0.8 to 0.1 in 2 years,... they may work technically under a guarantee but their performance can degrade. **Energy resources should be used directly**, for instance, daylighting through windows is far more efficient than through a photovoltaic-battery-fluorescent lamp conversion and one offsets the utility power by turning off the lights. **Negative capital** can be with cheaper refitting using more efficient, longer lasting, less powerful equipment. **Never retrofit** because the capital of an existing system it is replacing still has to be paid off during its useful life.... you **refit when it is economically justified**.

Sinking funds are amounts set aside to repay a debt at maturity or by schedule such as in public bonds and loans. During the interim, sinking funds are often reinvested to gain interest. In conservation, sinking funds can also be used to anticipate a future lifecycle cost or loan such as in effective refitting, and renovation. Batteries need replacement at their warranty period and hotels usually need to replace bedding, furnishings,.. every couple of years. **Energy contracts use this principle** where projected utility cost savings are used to replay a contractor who has installed conserving equipment at their expense. Financing here is often with high interest venture capital supplemented with government subsidies. Energy contracts are risky, rarely work out and the facility owner ends up with the bills along with replacing the poorly performing equipment. Performance based specifications were tried in the 70's by the National Bureau of Standards but were unsuccessful because prototypes failed to meet the specifications in testing and the parties involved could not work out who should fix and pay. Managing sinking funds is often very difficult, first in terms of monitoring the before and after utility costs, assessing baseline "savings", then to actually save the funds for the purpose.

Cost summary : Capital : Array panels **1.7 \$/Wsupply** under 1kW/m^2 sun **200 Wsupply 20%** efficient **340 \$/m²array 200 Wsupply/m²array** Batteries **1 \$/a-h** at 12V = **83 \$/kWh** allow oversize 185% for degradation , Grid-Tie Inverter **1700-2900\$** off-grid inverter only **800\$**, Controller **500\$** Ancillaries, cables,connectors.. **23% cost main systems**. Contractor installation cost **14% system cost**. Annual : Utility energy cost **0.13 \$/kWh** Grid-Tie fee ?? Insurance ?? Administrative fee : 10% within energy cost and likely for grid-tie. Sinking fund for maintenance to refit components : array **5%/y** battery bank **10%/y** inverter controller **10%/y** whole system needs replacement within **20** years. Rebates and tax credits are not included as the payback economics should be justified to both customer and taxpayer. Unfortunately governments are making poor technology investments in poor applications. Thus there is the need for this technology assessment which includes both viable performance and economic payback.

f. Analyzing scenarios :

Consider a **photovoltaic powered lighting scenario** for a **32x32 ft = 95 m² 1024 ft²** classroom in New Orleans **30°N** requiring less than **1.1 W/ft²** connected lighting and power demand **1.1x1024 = 1126 W** fluorescent lighting for an **8 hour 5 day** occupancy. The load energy becomes **1126x8 = 9.0 kWh/day**. Utility cost **1.126kW x 8h/day x 5/7day x 365day x 0.13\$/kWh = 2354kWhx0.13 = 306\$/y** The luminaire current **1126W / 120V = 9.4 a** Number of 32W T8 4ft lamps **1.1x0.9 = 1W/ft²x1024/32 = 30 lamps = 10 fixtures**, pendant 5 in 2 rows. From data generated in **d. e.** the diagram is reversed to size the equipment from the demand.



Powerwise the system is sized in kW for the voltage. Energywise it is sized in kWh for capacity a-h

Scenario 1. Grid-Tie and Array, no battery : sell back power from solar power collected
Peak solar power direct from array 30°N summer 1.06 A_A (m²) kW sum , 1.12 A_A (m²) kW win

$$1.06 A_A (m^2) kW = 1.126kW / 85\% \times 50\% \times 20\% = 13.3 kW \quad A_A = 13.3 / 1.06 = 12.5 m^2$$

array supply lighting demand inverter.circuit.array sun supply **minimum array area**

Summer solar energy direct from array to offset summer lighting demand.

$$A_A (m^2) \times 1046 W/m^2 \times 0.64 \times 0.84 \times 15.1 h \times 0.65 \times 183 kWh \times 20\% \times 50\% \times 85\%$$

$$\text{array mean } \Theta_{LatSum} + 10^\circ \quad k_\phi \quad k_{Alt} \quad T_{DaySum} \quad K_{SunSum} \text{ days available array circuit inverter}$$

$$= A_A (m^2) \times 85.852 kWh = 1.126 kW \times 8 h/day \times 5/7 day \times 183 day = 1177 kWh \text{ summer}$$

$$A_A = 1177 / 85.9 = 13.7 m^2 \text{ this offsets the summer utility energy for lighting}$$

Winter solar energy direct from array to offset winter lighting demand.

$$A_A (m^2) \times 1154 W/m^2 \times 0.64 \times 0.84 \times 10.9 h \times 0.50 \times 183 kWh \times 20\% \times 50\% \times 85\%$$

$$\text{array mean } \Theta_{LatWin} - 10^\circ \quad k_\phi \quad k_{Alt} \quad T_{DayWin} \quad K_{SunWin} \text{ days available array circuit inverter}$$

$$= A_A (m^2) \times 52.593 kWh = 1.126 kW \times 8 h/day \times 5/7 day \times 183 day = 1177 kWh \text{ winter}$$

$$A_A = 1177 / 52.6 = 22.4 m^2 \text{ this offsets the winter utility energy for lighting}$$

Now with a 25 m² array : Energy = 25x85.852 sum + 25x52.593 win = **3463 kWh**

which offsets 3463x0.13 = **450 \$/y** The most simplistic lifecycle costing without maintenance :

$$\text{Capital : } 25 \times 340 = \mathbf{8500\$} + \mathbf{2900\$} + \mathbf{500\$} + 23\% \times 11900 = \mathbf{2737\$} + 14\% \times 14637 = \mathbf{2049\$} = \mathbf{16686\$}$$

array inverter control ancillaries installation total

$$\text{Annual : } T_{Pay} [3463kWh \times 0.13\$/kWh = \mathbf{450 \$/y}]$$

Simple savings payback approach compared with conventional system $T_{Pay} = 16686/450 = \mathbf{37 \text{ year}}$

The rate becomes 16686 \$/ 2354 kWh/y x 20y = **0.35 \$/kWh** compared with a conventional utility **0.13 \$/kWh** a nearly 3 fold increase

Scenario 2. Grid-Tie Array, Battery : sell back power from solar power collected

Array 25 m² Power and energy as in Scenario 1 but we add the battery cost

Storage 2 day 1.126 kW/day x 8 h/day x 1day x 185% deg = **16.7 kWh** x 83\$/kWh = **1383 \$**

Capital : 25x340 = **8500\$** + **1383\$** + **2900\$** + **500\$** + 23% x 13283 = **3055\$** + 14% x 16338 = **2287\$** **18625\$**
 array battery inverter control ancillaries installation total

The rate becomes 18625 \$/ 2354 kWh/y x 20y = **0.40 \$/kWh** compared with a conventional utility

Scenario 3. Off-Grid. Array and Battery Standalone :

Peak solar power direct from array 30°N summer 1.06 A_A (m²) kW sum , 1.12 A_A (m²) kW win

1.06 A_A (m²) kW = 1.126kW/ 85% x 50% x 56% x 50% x 20% = 47.5 kW A_A = 47.5/1.06 = 44.8m²

array supply lighting demand inverter.circuit battery.array sun supply **minimum array area**

Winter (critical) solar energy direct from array to offset winter lighting demand.

A_A (m²) x 1154 W/m² x 0.64 x 0.84 x 10.9 h x 0.50 x 183 kWh x 20% x 50% x 85% x 50% x 56%

array mean $\Theta_{LatWin-10^0}$ k_φ k_{Alt} T_{DayWin} K_{SunWin} days available array circuit inverter battery
 = A_A (m²) x **14.726 kWh** = 1.126 kW x 8 h/day x 5/7 day x 183 day = 1177 kWh winter
 A_A = 1177 / 14.7 = **80 m²** this offsets the winter utility energy for lighting

Now with a 80 m² array : Energy = 80x85.852 sum + 80x52.593 win = **11080 kWh** of sun
 Standalone system takes in 11080 kWh sun to energize 2354 kWh lighting demand with energy lost to charge - discharge the batteries and turn off the system so the batteries do not overcharge particularly in summer.

Storage 3 day 1.126 kW/day x 8 h/day x 3day x 185% deg = **50 kWh** x 83\$/kWh = **4150 \$**

Capital : 80x340 = **27200\$** + **4150\$** + **800\$** + **500\$** + 23% x 32650 = **7510\$** + 14% x 40160 = **5622\$** **45782\$**
 array battery inverter control ancillaries installation total

This is a standalone cost of 45782\$/2354kWh/yx20y = **0.97 \$/kWh** for the life of the system compared with a conventional utility **0.13 \$/kWh** with over a **7** fold increase.

g. Assessing scenarios : for the described classroom and parametric criteria in track channel. Grid-Tie and Array, no battery, required an array **1/8** the floor area to establish a stable voltage and **1/7** to collect energy. To collect winter energy the array area is doubled **x2** for **1/4** the floor area.

A no-grid standalone with battery needs an array **4/5** the floor area at about **x6** fold a minimum grid area. By interpolation a third **Scenario 3. Grid-Tie with battery**, would be about **x4** fold a minimum distribution, **1/2** the floor area with the power divided **3:1 direct : battery** as the battery is only 0.5x0.56 = **28%** efficient compared with the direct.

Economics for all solar photovoltaic scenarios are unjustified. The simplest assessment is by the capital cost/annual energy used x economic life for the systems.

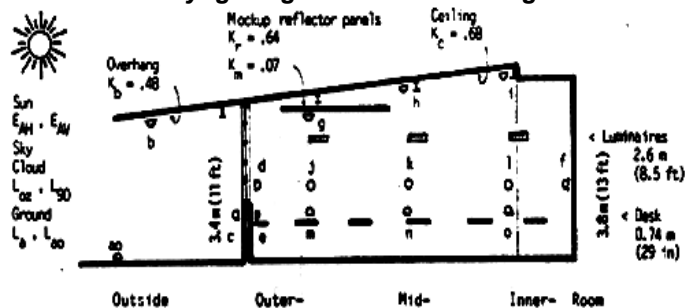
Conventional = **0.13 \$/kWh** Grid-Tie no battery = **0.35 \$/kWh**
 Grid-Tie Battery = **0.40 \$/kWh** Standalone Battery = **0.97 \$/kWh**

Payback periods are meaningless beyond the economic lifecycle of 20y, 37 years plus, and thus have no return on investment value. Degradation of arrays 50% in 20 years. There is no residual market value, in fact the owner has to pay for refitting or removal with little salvage value. Even solar thermal hot water heaters became obsolete and removed in California with lower gas prices. The whole idea of a utility is for an efficient economic distribution of resources. The best approach is to **conserve demands which lowers the need for supply**. In these scenarios conservation is in lowering the need for electric lighting through simply **direct sunshaded daylighting**. There is no need to change the energy mode from light to electricity back to light. Costs can be absorbed in a necessary refitting of systems at the end of their economic life and within the conventional construction.

Alternative Scenarios : always come up with a viable solution to the issues

Daylighting through sunshaded windows is a better scenario to **offset peak electric lighting**. This was ample for south facing classrooms with overhangs at El Roble Junior High, Claremont . An installed mirror over half the ceiling also improved the daylight penetration as illustrated

Sunshaded Daylighting with Mirror Ceiling



R38. Ceiling reflector panels for daylight penetration. John F. Halldane. Electric Power Research Institute Palo Alto. EPRI Research Project 2285-10, November 1985. El Roble High School, Claremont, CA

below. Architects these days do not seem to understand the basics; E-W facing windows need exterior vertical sunshading to keep out the heat and the blinding sun from your eyes, S facing windows need horizontal exterior sunshading overhangs to capture groundlight. Capital cost for mirror and windows with sunshading are part of the construction costs. **The lesson here is to use resources directly with the least of series inefficiencies and changes in energy modes.**

Acknowledgement :

It is my pleasure to have the opportunity to contribute to the Focus Symposium on Collaborative Decision-Support Systems through the Collaborative Agent Design Research Center in San Luis Obispo, California.

I truly thank my colleague Dr Jens Pohl of CADRC for this participation.

References :

1. **Technology Assessment: Applied Infocyber Scenarios.** John F. Halldane. 23rd International Conference on System Research, Informatics & Cybernetics, Baden-Baden, Germany. Aug 2011 focus symposium: ' Intelligent Information Management Systems ' Hosted by the International Institute for Advanced Studies in Systems Research and Cybernetics, IIAS. and the Collaborative Agent Design Research Center, CADRC, Cal Poly University, San Luis Obispo. **R136 G9**
2. **Collaborative matrix control, monitoring and management systems for infocyber channels, gates and tracks.** John F. Halldane. Inter Symp 2005. 18 International Conference on Systems Research, Informatics and Cybernetics : Collaborative Agent Design Research Center, CADRC, CalPoly University, San Luis Obispo. Baden-Baden. Germany. Aug8 2006 **R127.G1.**
3. **Design integration for minimal energy and cost.** John F. Halldane. Handbook. Elsevier Applied Science Publishers, UK, June 1989. Includes : power-energy models, lifecycle costing, the sameness criterion and the middle-third concept for statistics.
4. **Ceiling reflector panels for daylight penetration.** John F. Halldane. Electric Power Research Institute Palo Alto. EPRI Research Project 2285-10, November 1985. Comprehensive visual, photometric and economic evaluation with mock-up and model simulation of alternative designs. New portable sun-sky-cloud and model simulator. New impression-demonstration-interview method of enquiry. Response Demonstration Kit. Variation analysis with trends and excursions. Luminance distribution matrix. **R38. E5**
5. **Design evaluation in technology assessment: Illustrated by auditory impedance and sound distribution problems.** John Halldane. George Washington University, Monograph 5.12.72. Publication of the seminar presented to the Program of Policy Studies in Science and Technology, 3.9.72 Program sponsored by NASA.
6. **Developing simple energy saving methods for building standards.** John F. Halldane. Ministry of Sience, Technology & Environment , MOSTE, Vietnam. Implementing the Energy Efficiency Program in Vietnam. National Seminar 2 Energy Conservation & Efficiency Program, 10-11 March 1998, Paper Session 10. Co-organised: EDP Association for Energy Development & Planning, Netherlands. UN-ESCAP, Environment & Natural Resources Division, Bangkok. Royal Netherlands Embassy, Hanoi. UNDP, Hanoi. World Bank, Hanoi. **R101 , F11.**
7. **Performance characteristics of perimeter and core daylighting systems in low-rise commercial buildings.** John Halldane. Solar Energy Research Institute, SERI/TR-254-3092, Dec, 1986. Unpublished. Contains patentable invention disclosures. Defines daylight-collection-transposition-distribution systems **R41..**

8. **Comprehensive energy plan,** Federal Energy Administration. Report to Congress under Section 22, FEA act. Developed national conservation scenarios for Congress. Project Independence Evaluation System.-12.74 **R18.**

Semantic Transformation of Search Requests for Improving the Results of Web Search

Vladimir A. Fomichov and Anton V. Kirillov

Department of Innovations and Business

in the Sphere of Informational Technologies

Faculty of Business Informatics

National Research University “Higher School of Economics”

Kirpichnaya str. 33, 105679 Moscow, Russia

vfomichov@hse.ru and antonv.krillov@gmail.com

Abstract

The paper describes a new method of constructing semantic expansions of search requests for improving the results of Web search. This method is based on the theory of K-representations - a new theory of designing semantic-syntactic analyzers of natural language texts with the broad use of formal means for representing input, intermediary, and output data. The current version of the theory is set forth in a monograph published by Springer in 2010. The stated approach is implemented with the help of the Web programming language Java: an experimental search system AOS (Aspect Oriented Search) has been developed.

Keywords

Semantic transformation of search request; semantic representation; theory of K-representations; SK-languages; algorithm of semantic-syntactic analysis

Introduction

Every day the amount of information stored on the Internet is considerably increased. The format of presented information is heterogeneous, and its is unstructured; most often, the information is expressed by means of natural language (NL) – English, Russian, etc. Though there are known many approaches to the search for Web-based information (Kirillov, 2009; Halpin and Lavrenko, 2009; Fomichov, 2010), finding a solution to the following fundamental problem would be very important for the design of Web search systems – the calculation of the indicator of relevancy of the found document to the search request. In the course of studying this problem, a number of different approaches for recognizing a syntactic correspondence of a document to a search request: VSM (vector-space model), the functions BM25 and BM25F (taking into account the various weight factors of the words from a document), the functions Okapi, Ponte, the algorithm LCA and other. These approaches solve the problem of syntactic search, but a semantic correspondence of the found documents to the search request is not considered.

In order to solve this problem, several formats of meta-data describing semantic components of the documents have been developed, first of all, RDF, RDFS, OWL. Semantic description of a document provides the possibility to more exactly recognize its content and respectively the relevancy as concerns a search request. However, the documents very seldom include the meta-data of the kind, therefore the meta-data can be considered as a standard in the course of developing a Web-page. Since meta-information most often is inaccessible, the focus of the methods of finding the document relevance has shifted to the analysis of information stored in a natural language form.

During last years, many systems based on semantic analysis of the contents of requests and documents have been developed, in particular, SemSearch (Lei, Uren, and Motta, 2006), AquaLog (Bernstein, Kaufmann et al, 2005), Semantic Crystal (Bhagdev, Chapman et al, 2008).

Though there are numerous approaches to the search for information on the Internet, one observes the lack of the solutions combining the following possibilities:

- semantic-syntactic analysis of natural language search requests;
- typization of the requests;
- recognizing the objects of interest of a search request;
- the search for semantic equivalents of the objects of interest of a search request;
- finding the facts reflecting achieving a certain goal by an intelligent system;
- finding the evidence of the dynamics of certain sets (Management Boards of the firms, etc.).

The selection of just this collection of the possibilities is motivated by the following factors:

- a natural language interface allows for formulating the questions being of direct interest for the user but not forces the user to select a special combination of the words for successful syntactic search;
- the possibility to obtain the most complete collection of relevant information describing various aspects of the system's behavior, its state and achievements (or failures) of an intelligent system (including the organizations).

Central Ideas of the Proposed Solution

This paper proposes a solution optimizing the work of traditional search systems by means of semantic analysis and expanding the natural language input requests. Taking into account the calculating power of the biggest existing systems fulfilling the key words based search, it is proposed to shift the focus from the detailed semantic analysis and indexation of the content of electronic documents to the analysis of the inputted search requests and generation of a set of semantically expanded (adapted) requests that will be transmitted to a syntactic search system. The results of the search corresponding to each request from this semantically expanded set will be analyzed and compared with the aim of increasing semantic relevancy of the search results.

Example. Suppose that a user-businessman would like to get a certain information about the company X in order to consider the possibility of starting a collaboration with this company. In this connection, the questions about the achievements of the company during last year would be

quite natural. For instance, the user may ask the questions “What achievements did the company X have last year?” or “What failures did the company X have last year?”.

Both questions belong to the class of questions about the result of achieving a goal. Imagine that, as a result of the search, the user has received the information about the launch by the firm X of a new product or service Y. Correspondingly, the user would like to get the information about some characteristics of the product or service Y and, besides, about some distinguishing features of Y. The examples of the questions may be as follows: “What are the characteristics of the product Y?” and “What are the peculiarities of the product Y?”. The questions of this class will be called below *aspect-oriented questions*, and their processing will be considered in more detail.

Finally, having received the mentioned information, the user-businessman wants to get to know about the stability of the Board of Directors of the company X. For instance, he/she may formulate the question “What were the changes in the Board of Directors of the company X during last year?”.

With respect to the progress of voice interfaces and the computer means of synthesis and analysis of spoken speech, the process of looking for this information can be represented by the following dialogue:

User: “What achievements did the company X have last year?”.

System: “The company X launched the product Y, showed the benefit increase of 7%, and started a new office in Moscow”.

User: “What are the peculiarities of the product Y?”.

System: “High refusal stability and low price”.

User: “What are the distinctions of the product Y from the product Z?”.

System: “The product Y exceeds the product Z as concerns the following indicators:___”.

User: “What were the changes in the Board of Directors of this company during last year?”

System: “Peter Stein entered the Board of Directors”.

Thus, if a user wants to find information about a company, its achievements and failures, the launched products, various characteristics of the products, and about stability of its Board of Directors, then the complete process of search is covered by the proposed classes of questions and corresponding methods of search requests transformation. In this way, the speed, convenience, and relevancy of search will be increased.

The Method of Searching for Information of Interest

Let’s consider a method of looking for the information being of interest for the user under the framework on the proposed approach. A generalized algorithm consists of five main steps, two of them are unique for each of the considered types of questions.

Step 1. The inputted search request is analyzed for finding its type. It is necessary to distinguish the primary and secondary objects of interest of the search request W. Suppose that the request “What achievements did the company Intel have in the year 2010?”. Then primary object of interest is W1 = “achievements”, and secondary object of interest is W2 = “the company Intel”. The object W1 enables us to classify the search request W as an element of the class of questions about achieving a goal.

Step 2. After finding the type of the request it is possible to go to creating a set of secondary search requests generated by the request W, that is, to forming a semantic expansion of the inputted request. The construction of the semantically expanded set of requests is being fulfilled with the help of a knowledge base containing the information needed for forming new requests.

Step 3. As soon as the expanded set of requests has been formed, it is transmitted to the traditional search system, the latter returns a set of documents which syntactically correspond to the generated requests. Dependent on the preferences of the user, i.e. dependent of the user's behavior and selection of certain results of the search, the weights of the substitutions and the order of generating the requests (during the previous step) will be calculated.

Step 4. The documents received from the search system are analyzed and filtered with the help of a knowledge base (in order to calculate the number of occurrences of the indicators of interest in the document) and with the help of the indicators of documents' syntactic relevancy (the documents having the values of these indicators below a certain border will be excluded as non-relevant). The indicators will be understood as such natural language expressions that their occurrence in the text allows for judging about the correspondence of the document to the initial search request. First of all, the documents with the big amount of duplications will be considered. The reason is as follows: if a document more often occurs in the results of search proceeding from different requests, this document contains more indicators and, hence, contains more information corresponding to the initial request.

Step 5. The analyzed and filtered documents are then returned to the user.

Three classes of natural language questions are considered under the framework of the proposed approach, and the questions from these classes require certain speculations for constructing a semantically expanded set of search requests. These three classes of questions are (a) the questions concerning the achievement of a certain goal; (b) aspect-oriented questions, (c) the questions about the dynamics/stability of the sets (for instance, about stability of the Management Board of a certain company). Let's consider in more detail the methods of processing the search requests from the first class.

Processing of Questions about Achieving a Goal

We will say about the questions about achieving a goal in case of interrogative questions where one asks about information reflecting the results of functioning of an object, a system. In other words, these are *the questions about the achievements and failures*.

The success of functioning (or existing) of an object or a system is determined by achieving by the considered entity of the formulated goals. By a goal of a company we'll understand the final desirable result that is set in the process of planning and is regulated by the control functions. An example of the questions about achieving a goal is as follows: "What failures experienced the company Sun in the year 2010?".

For fulfilling a detailed analysis of questions about achieving a goal, we've selected, studied, and divided into several groups the goals associated with the activity of the enterprises. The examples of such goals are as follows: "The launch of a new product", "Starting a new office by a company", "The increase of benefit", "The absorption of a company".

The data of the kind should be stored in a special knowledge base, it will be called a *goal base*. This base is used for the generation of natural language expressions showing the availability in the documents of the information about success. The goal base is formed with the help of the theory of K-representations.

It is a new theory of designing semantic-syntactic analyzers of NL-texts with the use of formal means for representing input, intermediary, and output data is proposed (Fomichov 2010). This theory can be interpreted as powerful and flexible tool of designing the NL-interfaces to applied intelligent systems. The structure of this theory is as follows.

The *first basic constituent* of the theory of K-representations is the theory of SK-languages (standard knowledge languages). The kernel of the theory of SK-languages is a mathematical model describing a system of such 10 partial operations on structured meanings (SMs) of natural language texts (NL-texts) that, using primitive conceptual items as "blocks", we are able to build SMs of arbitrary NL-texts (including articles, textbooks, etc.) and arbitrary pieces of knowledge about the world.

The analysis of the scientific literature on artificial intelligence theory, mathematical and computational linguistics shows that today the class of SK-languages opens the broadest prospects for building semantic representations (SRs) of NL-texts (i.e., for representing meanings of NL-texts in a formal way).

The expressions of SK-languages will be called the K-strings. If T is an expression in natural language (NL) and a K-string E can be interpreted as a semantic representation T , then E will be called a K-representation (KR) of the expression T .

The *second basic constituent* of the theory of K-representations is a broadly applicable mathematical model of a linguistic database. The model describes the frames expressing the necessary conditions of the existence of semantic relations, in particular, in the word combinations of the following kinds: "Verbal form (verb, participle, gerund) + Preposition + Noun", "Verbal form + Noun", "Noun1 + Preposition + Noun2", "Noun1 + Noun2", "Number designation + Noun", "Attribute + Noun", "Interrogative word + Verb".

The *third basic constituent* of the theory of K-representations is a complex, strongly structured algorithm carrying out semantic-syntactic analysis of texts from some practically interesting sublanguages of NL. The algorithm *SemSynt1* transforms a NL-text in its semantic representation being a K-representation (Fomichov 2010). The input texts can be from the English, German, and Russian languages. That is why the algorithm *SemSynt1* is multilingual.

An important feature of this algorithm is that it doesn't construct any syntactic representation of the inputted NL-text but directly finds semantic relations between text units. The other

distinguished feature is that a complicated algorithm is completely described with the help of formal means, that is why it is problem independent and doesn't depend on a programming system. The algorithm is implemented in the programming language PYTHON.

The formation of a goal base is semi-automated. The first step consists in processing a special representation of a goal with the help of the algorithm *SemSynt1* described in (Fomichov 2010). For instance, the knowledge engineer inputs the sentence $S1 = \text{"\#The company X\# absorbs the company Y"}$. Here the marker # is used for distinguishing such entity that its collection of goals includes the goal described in the considered sentence.

As a result of semantic interpretation of the sentence $S1$, the following K-representation *Semrepr1* of $S1$ will be constructed:

$$\begin{aligned} & (Situation(e1, absorption1 * (Agent2, certn company1 *(Name1, X) : z1) \\ & (Dependent-org, certn company1 *(Name1, Y) : z2)) \wedge (z1 \equiv Ob-intr)), \end{aligned}$$

where the variable *Ob-intr* is interpreted as the designation of an object of interest in the future search request.

Then the knowledge engineer constructs an expanded expression

$$\begin{aligned} & <(Situation(e1, absorption1 * (Agent2, certn company1 *(Name1, X) : z1) \\ & (Dependent-org, certn company1 *(Name1, Y) : z2)) \wedge (z1 \equiv Ob-intr)), +1>, \end{aligned}$$

where the symbol +1 indicates that the truth of the sentence $S1$ reflects the achievement of a goal of the company X.

The K-string *Semrepr1* is used for constructing the pattern

$$\{org\} [absorption1] (verb) \{org\}.$$

The success of comparing this pattern with a document will be achieved in case when this document includes a distributed word combination $A B C$, where A and C are the lexical units associated with arbitrary concretizations of the semantic unit *org* (organization), B is a lexical unit associated with the semantic item *absorption1*.

The descriptions of the achievements and failures (let's call them the facts) are stored in the goal base and are used for the generation of word combinations being the indicators of the document fragments mentioning these achievements or failures. Consider in more detail a method of transforming a fact into a word combination – indicator.

The construction is being fulfilled with the help of the transformation rules being unique for each fact. A transformation rule indicates the order of the words in the word combination and the forms of combinations. These combinations will enable a traditional search system realizing the search on key words to find all documents mentioning the relevant facts. The collection of the documents returned by a search system will be analyzed from the standpoint of calculating the quantity of occurrences of various combinations – indicators, that is, the indicators of a reference in the document to a fact. The relevance of a document will be determined, firstly, by the quantity of occurrences of various facts and secondly – on the rating of a document calculated in

accordance with the algorithm PageRank. The documents sorted with respect to its relevance to the initial search request will be transmitted to the user.

The stated approach is implemented with the help of the Web programming language Java: an experimental search system AOS (Aspect Oriented Search) has been developed. Now the system AOS is being tested.

References

- [1] Bernstein, A., Kaufmann, E., A. Gohring, A. and C. Kiefer (2005); Querying Ontologies: A Controlled English Interface for End-users. In 4th International Semantic Web Conference (ISWC 2005), pages 112– 126, November 2005 (pp. 112-126)
- [2] Bhagdev, R., Chapman, S., Ciravegna, F., Lanfranchi, V. and D. Petrelli (2008); Hybrid search: Effectively combining keywords and semantic searches. In The Semantic Web: Research and Applications; Springer, Berlin / Heidelberg (pp. 554–568)
- [3] Fomichov, V.A. (2010); Semantics-Oriented Natural Language Processing: Mathematical Models and Algorithms; Springer, New York, Dordrecht, Heidelberg, London (354 pp)
- [4] Halpin, H. and Lavrenko, V. (2009); Relevance Feedback Between Hypertext and Semantic Search; Proc. International Conference WWW2009 (April 20-24, 2009, Madrid, Spain).
- [5] Kirillov A. (2009); Search Systems: Components, Logic, and Methods of Ranging; Business-informatics (Moscow), No. 4 (10) (pp. 51-59)
- [6] Lei, Y., Uren, V. and E. Motta (2006); Semsearch: A search engine for the semantic web. In Proc. 5th International Conference on Knowledge Engineering and Knowledge Management Managing Knowledge in a World of Networks, Lect. Notes in Comp. Sci., Springer, Podebrady, Czech Republic (pp. 238-245).

An Agent Facilitated Design Conversation System for Facilitating the Designer in Creative Thinking in Architectural Design

GEETA ARJUN, JIM PLUME
Faculty of The Built Environment
University of New South Wales NSW 2052
Australia
geetalarjun@yahoo.com, j.plume@unsw.edu.au

Abstract. The paper discusses a current research that investigates if a computer aided conversation system can be created to support the human thought process in the early stages of architectural design. It argues that design conversations are an essential premise for designing, especially at the early stages, when the designer has to brainstorm ideas to generate creative conceptual solution-conjectures. The paper also argues that design knowledge is mainly dependent on a designer's experiences. But experiential knowledge, stored in the long term memory is difficult to recall. Based on these arguments, an agent-based knowledge system, *Design Thinker*, is designed to allow for an efficient design conversation that triggers the experiential memory of the designer for recalling and associating the right experiences. It is also designed to enhance and add to the existing design knowledge of the designer by enabling them to view the knowledge through different perspective or domain lenses. The paper describes the conceptual structure of the knowledgebase used in the prototype followed with a brief overview of the empirical study.

Introduction

The early stage of architectural design is often the most creative period in the design process. It is in this stage that the designer begins to formulate ideas for developing conceptual solution conjectures. A two-way interaction begins between the designer and the developing design, leading to further new ideas for the design situation. Donald Schön (1995) in his seminal work, compares this dynamic, cyclic and unfolding nature of the design process as the designer having a *reflective conversation* with the design situation. In other words, a designer talks to their drawings and the drawing speaks back as if showing a new perspective of the design situation.

A meaningful design conversation is based on the knowledge of the design conversationalists. In this study, the focus is on the experiential knowledge of the designer. Generally in architectural practices, a team of designers discuss ideas to arrive at a suitable design concept or solution for a specific design situation. Each designer brings with them knowledge in terms of their design experiences. This experiential knowledge is shared by the designers and in turn triggers the generation of new ideas. The experiential knowledge is not necessarily through work experience alone, but is collected over time through the designer's exposure to various design elements like design precedents, pictures and also elements from other related design fields like fashion, photography etc. This experiential knowledge is stored in the long term memory of the designer and is generally very difficult to recall unless triggered. Lawson (2004) highlights the role of conversations in this respect, where one idea triggers another, apparently remote from it. This also suggests the significance of words as most conversations involve words to communicate thoughts and ideas.

Based on the arguments, it is hypothesized that a computer aided conversation system can aid the designer in triggering ideas during the early stages of the design process.

The aim of this study is to test the hypothesis. To achieve this, there is a need to gain insight into the state-of-art computational technologies in knowledgebase systems and artificial intelligence. Based on this study, a design conversation prototype, *Design Thinker*, is proposed and implemented as a dialogue between the user and domain-specific computational agents.

In this paper, the focus is on the design and architecture of the prototype, *Design Thinker*. Section 1 presents the arguments for the hypothesis. Section 2 presents the design of the implemented prototype, with an emphasis on the structural representation of the knowledgebase and the conceptual design of domain agents. In section 3, the empirical study used to test the performance of the system is described briefly, followed by the results and conclusions drawn from the study in section 4.

Design Conversations

In the early stages of the design process, architects usually begin with a pre-briefing session with the clients to gain an understanding of the basics of the design problem. This is followed by interpreting and often developing the design brief, understanding the requirements, visiting the site and holding further meetings with the clients. The design thinking period begins from the pre-briefing stage and continues throughout the design process (Luck and McDonnell, 2006).

Architectural design, by nature, is also a response considerations that span a wide range of domains such as aesthetic, functional, material and ecological. These are often inconsistent, but are nevertheless brought together through architectural design in a novel way (Haapasalo, 2000). It is through a reflective conversation with the design situation, generally informed by active conversation with experts from many different domains, that the designers display a capability for integration, evaluation and synthesis of complex ideas from the different domains of architectural design. Conversation is an intrinsic part of human nature. The dictionary defines conversation as an informal talk with somebody about opinions, ideas, feelings or everyday matters. Apart from the reflective nature of design conversations, Loke (1997) also identifies the

generative nature of design conversations since, in a design conversation, not only known information is being transferred between the conversationalists, but new information and insights are discovered which neither conversationalist would have known. The reflective and generative aspects of design conversations highlight the significance of design knowledge gained through a design conversation.

Design Knowledge

Design is essentially a collaborative process. As stated above, the early stages of design mark the need for an outstanding capability for integration, evaluation and synthesis of concepts. For these reasons, it is generally common for architectural practices to employ design teams rather than individual project designers. The former provide a rich collective experience from different domains. So how does this experiential knowledge build up? Lawson (2004) highlights the use of the 'precedent'. He identifies the precedents as a wide variety of knowledge related to design that gets stored in the designer's 'experiential' memory. Such precedents are described as employed solutions by the designer or other famous designers, buildings, landscapes, towns seen during travel and even through media images. It can also include elements from other design fields like fashion, photography, products and others. All this exposure is said to build a designer's knowledge, especially the experiential knowledge which the designer can draw upon in future design problem-solving. Comparative studies between experts and novice designers clearly indicate the use of experiential knowledge as a vital factor in the design process (Cross, 2006, Goker, 1997). Cross (2006) identifies the development of design ability to be through 'experience'. In his comparative observations between experts and novices, he argues that experienced designers are able to draw on their knowledge of previous experiences in their field of design using solution- conjectures for a rapid exploration of the problem. He states, (Cross, 2006, pg 26)- 'They (experienced designers) use early solution attempts as experiments to help identify relevant information about the problem. In comparison, novice designers often become bogged down in attempts to understand the problem before they start generating solutions'.

Every designer, including the novice, has a certain level of design experience gained from childhood, but there still seem to be some designers who can recall their experiences to a particular problem efficiently and are seen to have better design ability.

This highlights two needs – one of informing the designer and adding to their experiences while designing and the other of aiding the designer by triggering their experiential knowledge from their long term memory for solution conjectures in exploring the problem.

The essence of any conversation is the sharing of thoughts and ideas through 'words'. Words in design, rather than pictures are seen to be more useful triggers for design knowledge. Loke (1997) comments that a picture triggers the sensual experiences, but these experiences are hard to remember at will. Words label experiences whereby the latter acquire more meaning and are easily recalled when needed. Words in combination with pictures can provide a powerful tool for conducting design conversations.

The Prototype

Taking the analogy of Schon's reflective conversation theory, a prototype, Design Thinker, is proposed as a dialogue between the user and domain-specific computational agents. The dialogue is intended to trigger the experiential memory of the user and associate significant experiences from different domains of the design problem to stimulate creative thinking. This model represents 4 different design activities theorized by Schon- *naming*, *framing*, *moving* and *reflecting* (Schon, 1995, Valkenburg and Dorst, 1998). The user begins the system by naming a design consideration. The *framing* agent frames alternative ideas for the design consideration from the system knowledgebase. Once an idea is selected, it marks the beginning of the design conversation. The moving and reflecting phase involve an interaction between the user and *domain* agents in which domain agents provide their individual perspectives in return to a user-selected response and the process continues. These domain perspectives are drawn from a filtered view of the overall knowledgebase, derived from an understanding of each domain agent's area of focus or interest.

THE KNOWLEDGE-BASE

The agent system is based on a *Blackboard Architecture* in which a blackboard forms a temporary database. At the core of the agent system is the knowledgebase (or ontology) that forms the basis of agent reasoning. The ontology for the system is adapted from the knowledgebase of the book, 'The Metapolis Dictionary of Advanced Architecture.' This dictionary defines the practice of architecture in a contemporary perspective by providing definitions and meanings to a wide variety of terms that are associated with what has come to be called 'Advanced Architecture'. This ontological dictionary serves as a fine example of the semantics and associations between words in a particular domain of architectural design, making an appropriate resource upon which to build this prototype system. The ontology is structured in the book, as shown in Figure 1, providing a straightforward structure for the knowledgebase that lies at the heart of *Design Thinker*.

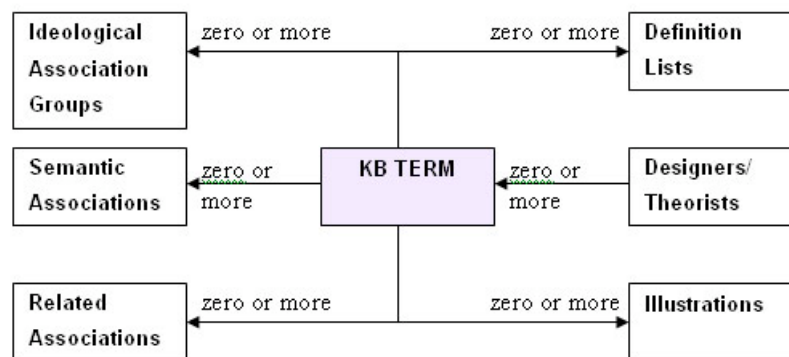


Fig 1. Structure of the knowledgebase as presented in the 'The Metapolis Dictionary of Advanced Architecture'

Each KnowledgeBase Term (KB Term), which is a main term in the knowledgebase related to architectural design has 3 sets of associations (with other KB terms: Ideological, Semantic and

Related) as well as supplied textual definitions and illustrations. In addition, there is a list of designers and theorists who in turn are associated with a list of key terms.

The knowledgebase segments are described in more detail as follows:

1. *Ideological associations*

A key term is associated ideologically to zero or more groups of analogical associations consisting of *KB Terms*. This association is the first point of consultation/lookup for the domain agents.

2. *Semantic associations*

Each *KB Term* may or may not be related to a semantic association. A semantic association is used when the *KB Term* is close enough to be explained through the definition of another term. The semantic association is the second point of consultation for the domain agents.

3. *Related associations*

Each *KB Term* is provided with a list of links, that is, a series of words related to that term. These terms provide more information on the current *KB Term*. The related associations form the third point of consultation for the domain agents.

4. *Definitions*

The definitions following each *KB Term* provide an understanding of the *KB Term* from different authors' perspectives. The definitions are the fourth point of consultation for the domain agents.

5. *Designers/ theorists*

The book provides an index of architects, designers, critics, engineers, philosophers with a list of *KB Terms* associated with them. It is a tool to understand what kind of architecture, which position or which theme everyone deals with. The list of designers' et al are the fifth point of consultation for the domain agents.

6. *Illustrations*

Each *KB Term* may have zero or more referenced illustrations along with the definitions to facilitate quick consultation and explanation of the terms. Most of the illustrations are from architectural projects and show the characteristics identified in the definition of the corresponding *KB Term*. These form the last point of consultation for the domain agents.

THE FRAMING AGENT

A user enters a word as a *Design Consideration*, which is picked up by the framing agent. The task of the framing agent is to identify *Candidate Ideas* from the knowledgebase that are significant to the *Design Consideration*.

The framing agent carries out a text search for the *Design Consideration* in the list of *KB Term Definitions* segment of the knowledgebase. When it finds a match, the corresponding *KB Term* is added to the list of *Candidate Ideas*. A list of *Candidate Ideas* is determined. Each *Candidate Idea* is assigned a score and the first 3 ideas with the highest score are presented to the user as

Ideas. If the user is not satisfied, he/she can prompt the framing agent to provide further ideas in groups of 3.

THE DOMAIN AGENTS

As the name suggests, domain agents belong to a particular domain and respond to the *Focus Term* on the blackboard based on their own *View* of the knowledgebase. This *View* is a subset of the entire structure of the main knowledgebase. The user can also create additional domain agents which can be added to the system.

SCORING OF CANDIDATE RESPONSES

When an *Idea* (returned by the Framing Agent) is activated by the user to initiate a conversation, it is picked up by each domain agent and becomes the *Focus Term*. Each domain agent searches for *Candidate Responses* in each KB segment of its *View*. It compiles a list of *Candidate Responses* from each KB segment and scores them. A conceptual basis for the choice of *Candidate Responses* from each KB segment is as follows:

1. Ideological associations

Each key term is associated ideologically to a group of zero or more analogical associations. In response to a *Focus Term*, the domain agent determines a list of *Candidate Responses* that are most relevant. For a term to qualify as a *Candidate Response*, it must contain the *Focus Term* in one of its association groups. Each *Candidate Response* is assigned a score reflecting its importance.

2. Semantic associations

In response to a *Focus Term*, the domain agent determines a list of relevant *Candidate Responses* that are most relevant as semantic associations. For a term to qualify as the *Candidate Response*, it must contain the *Focus Term* as one of its semantic associations. Each *Candidate Response* is assigned a score reflecting its importance.

3. Related associations

In response to a *Focus Term*, the domain agent determines a list of *Candidate Responses* that are most relevant as related associations. The agent locates the *Focus Term* in the *KB Terms* of its knowledgebase. If it finds a match, the related associations for that *KB Term* become the *Candidate Responses*. Each *Candidate Response* is assigned a score reflecting its importance.

4. Designers/ theorists

In response to a *Focus Term*, the domain agent determines a list of relevant *Candidate Responses* that are designer names. The agent locates the *Focus Term* in each designer group containing a list of *KB Terms*. If it finds a match, the corresponding designer names become the *Candidate Responses*. Each *Candidate Response* is assigned a score reflecting its importance.

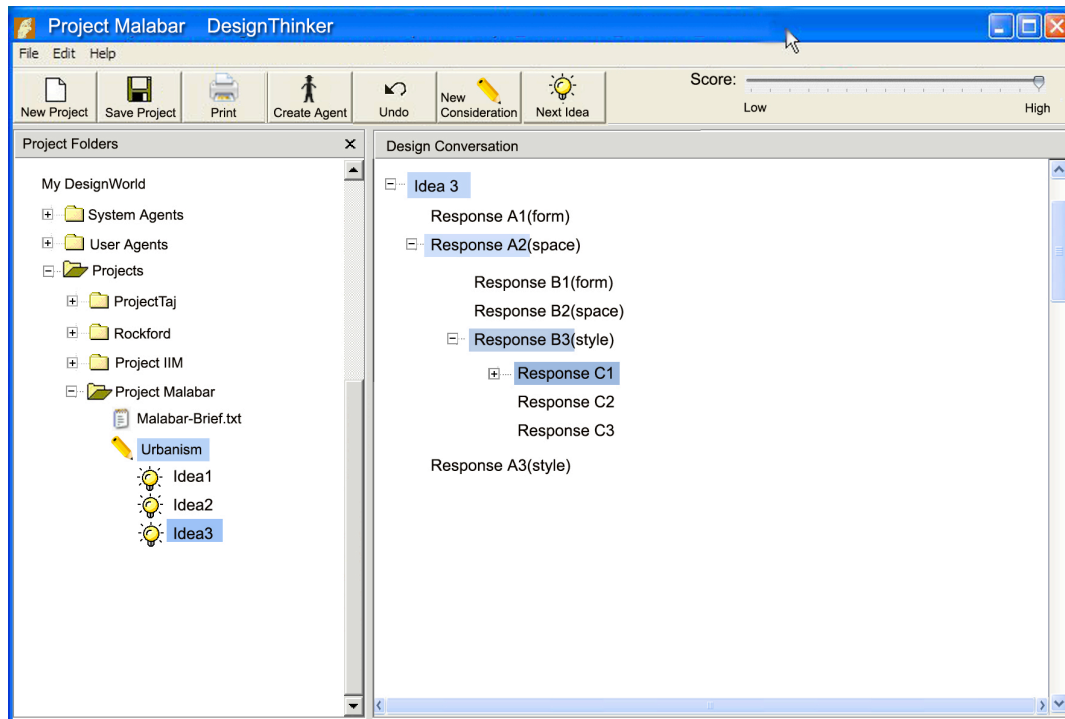


Fig 2.Design Thinker as an Explorer Interface

THE CONVERSATION WITH DESIGN THINKER

Design Thinker has been implemented as a design conversation system to facilitate the designer in the generation of ideas at the early stages of the design process. Figure 2 shows the working interface of Design Thinker. For a particular design project, a designer enters the project brief and a Design Consideration that is an issue related to the project and of the designer's interest. When a Design Consideration is entered into the system, the framing agent is activated, who prompts a set of three Candidate Ideas from the overall knowledgebase as per the scoring system. The designer can choose to interact with the framing agent for more relevant ideas by changing the score on a displayed score bar on the interface and clicking the 'Next Idea' button.. At this stage, the designer can also change the Design Consideration, to be prompted for more ideas by the framing agent. Once an idea is selected by the designer, it is picked up by each domain agent in the system and becomes the Focus Term. Prior, to this the designer can also create new domain agents or choose the domain agents from the available list to make them active for a design conversation. Each domain agent returns one Response to the interface The user can continue the conversation by selecting a response through a double click on the word. The selected response is highlighted on the interface and is transferred to the blackboard as a *Focus Term*. In return, the domain agents provide new *Responses* from their *Views* based on previous scoring methods and the conversation continues. On right-clicking a response, a user can choose to view the detailed definition and illustrations for the respective response to facilitate an explanation for the term.

Empirical Study

An experiment was conducted with ten designers with varying degree of professional experience. Only one participant participated in the experiment at any one time. Two design tasks were set for the experiment, one to be completed without using Design Thinker and one using Design Thinker. In both design settings, the participants had to express their thoughts aloud, right from the beginning to the end of the design process. The participants were given A4 sheets of paper for sketching their design ideas and both the sessions were recorded on video. The video recordings enabled the researcher to go through each session for further detailed analysis. The working of Design Thinker was explained to the participants before beginning the second design task. At the end of each design session, each participant mapped their design ideas from the beginning to the end of the design process in a concept map software program called 'CMaps'. These maps provided a cognitive view of the designer's design process. At the end of both the design tasks, each participant completed a questionnaire followed with an interview that allowed each participant to express their viewpoint on using Design Thinker during the early stages of the design process. Both the sessions were analysed using protocol analysis, segmenting and coding the activities based on Schon's paradigm of naming, framing, moving and reflecting. In addition, the design outcomes for both protocols were rated by external judges.

Results and Conclusions

The results indicated that for around seventy percent of the designers, Design Thinker did trigger their memory for ideas and solution conjectures in solving the design problem. Sixty percent of the designers were able to recall their experiential memory for ideas and think in parallel on several design issues. The ratings by external judges also demonstrated that the use of Design Thinker in the design process did trigger idea generation in some designers and had an impact on the quality of their solutions. An interesting aspect of the results was that participants with lesser experience benefited and appreciated the system more than designers with a higher level of experience. Based on this result, the study also indicates that Design Thinker could be a useful pedagogic tool in the education of architectural design.

References

- Cross (2006), *Designerly ways of knowing*, Springer, London
- Göker M H (1997) The Effects of Experience During Design Problem Solving *Design Studies* 18(4): 405-426
- Haapasalo, H. (2000), *Creative Computer Aided Architectural Design: An internal approach to the design process*, Academic Dissertation, Faculty of Technology, University of Oulu, Oulu.
- Lawson, B. R (2004), *What Designers Know*, Architectural Press, Oxford
- Lawson, B. R and Loke, S. M. (1997), Computers, words and pictures, *Design Studies*, vol.18, pp171-184
- Loke. S. M. (1997), 'Sketching with words ', PhD thesis, University of Sheffield, United Kingdom

Luck and McDonnell (2006), 'Architect and user interaction: the spoken representation of form and functional meaning in early design conversations ', *Design Studies*, Vol 27, No.2 pp 141 - 166

Schön, D.: 1995, *Reflective Practitioner: how professionals think in action*, New edn, Aldershot Press, England

Valkenburg and Dorst (1998), 'The reflective practice of design teams', *Design Studies*, Vol 19, pp 249 -271

GLOSSARY OF TERMS

Blackboard: An in-built database of user-selected *Focus Terms* respectively. The blackboard functions as an internal mechanism.

Candidate Ideas (ci): A set of potential ideas identified from the knowledgebase by the framing agent in return to a *Design Consideration*.

Candidate Response: A set of potential responses provided by the domain agents in return to a *Focus Term* on the blackboard.

Design Consideration (dc): A design key term for the current project entered in by a user that activates the framing agent.

Focus term (ft): A term selected from Responses or Ideas and placed on the blackboard.

Ideas (i): Ideas are a set of terms from an ordered list of *Candidate Ideas* provided by the framing agent that are identified and placed on the interface in groups of three.

Ideological Association (IA): The segment of knowledgebase containing the 'ideological dictionary' with its set of analogical groups for a key term.

KB Term (KBt): The main terms in the knowledgebase and agent views supplied with textual definitions and illustrations.

Knowledgebase (kb): The ontology or knowledgebase used for agent communication. In this prototype, the knowledgebase is adapted from the book, 'The Metapolis Dictionary of Advanced Architecture'.

Related Association (RA): The segment of knowledgebase listing the related associations for a *KB Term*.

Semantic Association (SA): The segment of knowledgebase listing the semantic associations for a *KB Term*.

View: A subset of the knowledgebase extracted by the domain agents by matching their characteristics to words in the *KB Term* definition segment of the knowledgebase.

ICODES: A Load-Planning System that Demonstrates the Value of Ontologies in the Realm of Logistical Command and Control (C2)

Kym J. Pohl, Collaborative Agent Design Research Center, Cal Poly, San Luis Obispo, CA
Peter Morosoff, Electronic Mapping Systems (E-MAPS) Inc., Fairfax, VA

Over the past decade the Collaborative Agent Design Research Center (CADRC) at California Polytechnic State University (Cal Poly, San Luis Obispo) and its sustaining sponsor CDM Technologies, Inc. (CDM) have developed a suite of information-centric software tools in support of military deployment and distribution processes. At the heart of each of these tools are expressive context models or ontologies that are partnered with select communities of software agents capable of reasoning about domain-specific information and concepts to provide their user communities with meaningful decision-support. Collectively these tools represent an evolving suite of adaptive Knowledge Management Enterprise Services (KMES) that can be readily configured into a net-centric, Service-Oriented Architecture (SOA) based planning and decision-support toolset for a particular application domain. As a set of KMES tools the Integrated Computerized Deployment System (ICODES) is configured to support the movement of supplies in the military deployment and sustainment operational domain. The application focus is conveyance load-planning, including the staging of cargo in marshalling yards, assembly areas, and rail heads.

ICODES has been a program of record (POR) successfully employed by the United States (US) Department of Defense (DoD) since 1997. Incorporation of progressive technologies such as ontological representation and agent-based analysis was a reaction to (1) the Army's experience with the movement of supplies by ship in support of Operation Desert Storm showing that the traditional manual approach to load-planning impeded operations due to the unanticipated changes and subsequent problems that inevitably arise and (2) DoD's realization that it was increasingly infeasible to employ the number of people needed to continue using a substantially manual approach to load-planning operations. These operational and fiscal realities forced DoD to find ways to use progressive computer-based technologies to reduce costs and improve operational effectiveness.

Command and Control (C2)

ICODES was developed as an information-oriented, agent-based system in direct response to the complexities inherent in military load-planning. To effectively apply ontology-based technology to C2, it is important to keep in mind its primary definition and set of objectives. The authoritative definition of C2 is:

"The exercise of authority and direction by a properly designated commander over assigned and attached forces in the accomplishment of the mission. Command and control functions are performed through an arrangement of personnel, equipment, communications, facilities, and procedures employed by a commander in planning, directing, coordinating, and controlling forces and operations in the accomplishment of

*the mission.” (Joint Publication [JP] 1, *Doctrine for the Armed Forces of the United States*)*

Stated simply, C2 is commanders: (1) learning what they need to know to make good decisions that lead to the successful accomplishment of their missions; (2) making these decisions; (3) issuing directions; and, (4) supervising the execution of such directions. Warfare is inherently non-deterministic and therefore the ability of the military decision maker (or commander) to have access to timely, accurate, and actionable information is absolutely critical. Prior to ICODES, the data and information used in planning and executing a shipload was employed almost entirely by shipload specialists. From inception, the development of ICODES focused on the user’s business processes, the very nature of information relevant to C2 operations, and translating the large volumes of data into meaningful information that users need. Equipped with ICODES, shipload specialists found that they could process data and information to produce valuable information that commanders could easily understand and quickly exploit to make important decisions. As a result of the situation awareness and related information provided by ICODES, commanders have seen a significant reduction in uncertainty and therefore improved the quality of their decisions.

Throughout its use as a DoD load-planning resource, operational experience with ICODES showed that an information-based approach provides commanders and their staffs with a set of C2 support-tools that can process very large quantities of data to produce accurate information that is presented in easy-to-understand displays. Such transformation of what is typically an overwhelming sea of data into relevant, actionable information is critical as the US reduces the size of its armed forces.

Load-Planning as a Complex Problem

The rapid deployment of military assets from the US to overseas locations is a complex undertaking. It involves the movement of large numbers of tracked and wheeled vehicles, weapon systems, ammunition, power generating and communication facilities, fuel, food supplies, and other equipment and goods, from military bases to the area(s) of operation. Several modes of transportation are typically involved. Depending on the location of the military base the assets are preferably moved by road to the nearest railhead, from where they are loaded onto railcars for transportation to the appropriate air or ocean port of embarkation.

Alternatively, if rail transportation is not an option, all of the cargo must be shepherded through the public road corridor from the base to the port. At the port of embarkation the assets are briefly assembled in staging areas and then loaded onto aircraft or vessels for shipment. Points of debarkation may vary widely from a commercial air or ocean port with fairly good facilities to a secure airfield in the theater or an amphibious landing on a hostile shoreline under fire. Once the cargo has been disembarked in or near the theater it, must be transported to its final destination by road, rail, air, or barge. In many cases this becomes an inter-modal affair with the need for frequent re-planning due to changes in priority or as routes in the theater become temporarily unavailable due to inclement weather or enemy activities.

Speed and in-transit visibility are of the essence (Figure 1). The total time required for the loading and unloading of the conveyance is a critical factor and largely determined by the quality of the load-plan. Ship load-planning, for example, has many of the characteristics of a complex problem situation (Figure 2). First, there are continuous information changes. The vessel that

arrives at the port may not be the vessel that was expected and that has been planned for. This means that the existing load-plan is no longer applicable and a new plan has to be developed. Similarly, last minute cargo changes or inoperative lifting equipment may require the existing plan to be modified or completely revised. Second, there are several complex interrelationships. The cargo on any one ship may be destined for several ports of debarkation, requiring careful consideration of loading and unloading sequences. However, these sequences must take into account unloading priorities that may be dictated largely by tactical mission plans. In addition, the placement of individual cargo items on board the ship is subject to hazardous material regulations and practices. These regulations are voluminous and complex in themselves. At times they are subject to interpretation, based on past experience and detailed knowledge of maritime risks and practices. Finally, the trim and stability characteristics of the ship must be observed throughout the planning process. This includes listing, draft and deck stress limitations.

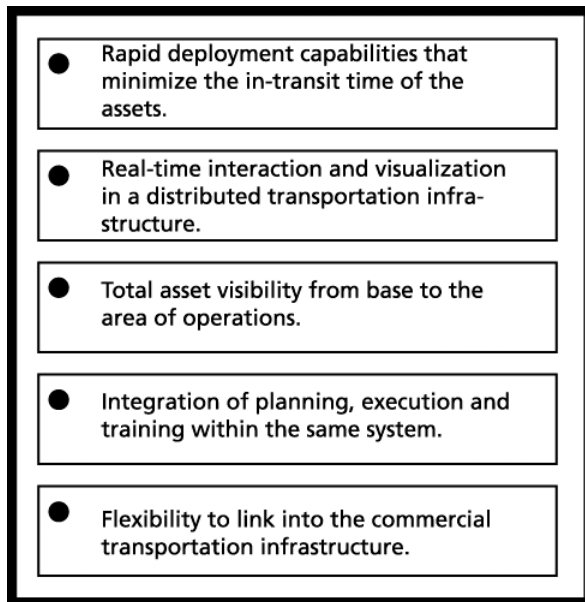


Figure 1: Military deployment objectives

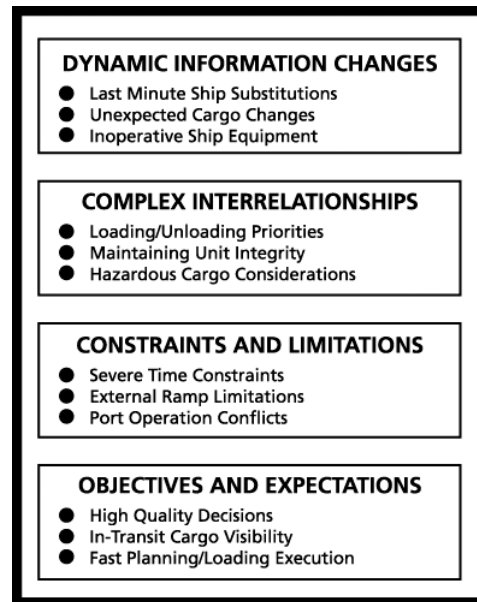


Figure 2: Complexity of ship load-planning

Third, there are many loading and unloading constraints. Some of these constraints are static and others are dynamic in nature. For example, depending on the regional location of an ocean port external ship ramps may not be operable under certain tide conditions, or an airfield may be able to accommodate only a small number of aircraft concurrently on the ground for loading purposes. Local traffic conditions, such as peak hour commuter traffic and rail crossings, may seriously impact the movement of cargo into staging areas or from staging areas to the pier or aircraft loading area. While these constraints are compounded whenever loading operations occur concurrently, the general complexity of the load-planning problem is exacerbated by the number of parties involved. Each of these parties plays an important role in the success of the operation, but may have quite different objectives. Certainly, the objectives of the commercial stevedore crews that may be under contract to carry out the actual loading tasks are likely to differ markedly from the prevailing military objectives that include rapid loading and unloading operations, unit integrity, load density, documentation accuracy, and security.

The ICODES Solution

To effectively address the complexities inherent in military load-planning it is necessary to have a solution that incorporates intelligent software capable of analyzing the large amounts of data, the concepts critical to the load-planning activity, and most importantly the extensive relationships that bind these components together.. Within such a decision-support facility, expressive information models, or ontologies, provide the context necessary for in-depth analysis to occur.

ICODES is an example of a new generation of *information-centric* military decision-support software tools that feature expert agents with automatic reasoning and analysis capabilities. This is made possible by an internal virtual representation (i.e., domain model or ontology) of the load-planning environment, in terms of conveyance and cargo characteristics and the complex relationships that constitute the context within which load-planning operations are performed. ICODES agents employ these rich ontologies to monitor the principal determinants of cargo loading, including: the placement and segregation requirements for hazardous cargo items; the trim and stability requirements of the conveyance; the accessibility of stow areas; the correct placement of cargo items in respect to restricted areas and inter-cargo spacing tolerances; and, the accuracy of cargo characteristics (e.g., dimensions, weight, type, and identification codes) relative to standard cargo libraries and associated reference tables.

Expert Agent Capabilities

There are many definitions of software agents in the literature (Wooldridge and Jennings 1995; Bradshaw 1997). To the authors, a software agent in its simplest form is a software module (i.e., service) that is capable of communicating with other software modules or human agents to facilitate some action. However, at this level of definition an agent is not necessarily intelligent. An intelligent agent would need to communicate using a common language (such as the ontology represented by the Semantic Network in ICODES) to support reasoning capabilities. In addition, an agent may have deep information and expert skills within a narrow domain and would then be referred to as a knowledge-based agent that has the ability to act on its own initiative. Such agents typically collaborate with other software and human agents to accomplish goals, and use local information to manage local resources.

The expert agents in ICODES are designed to assist the load-planner in the knowledge domains of hazardous material, trim and stability of the ship, cargo access paths, cargo attribute verification, and the actual placement of cargo in stow areas. The agents do not communicate directly with each other, but are totally decoupled. In fact, they do not know about each others existence. They collaborate indirectly as clients through a subscription service that allows them to post interests to data changes within the context provided by the ontology.

When the user is developing a load-plan while operating in *User Stow mode*, where users manually place cargo items within the various stow areas, the agents continually analyze the evolving load-plan and alerting the user to any violations or concerns that may arise. Agents communicate with users through computer-monitor displays by turning the surround of the appropriate agent status window red. By selecting the highlighted agent icon, the user can interact directly with the agent and obtain an explanation of the violation and related implications.

When users operate in what is referred to as *Assisted Stow mode*, they perform an initial configuration of preferences and restrictions used to drive the automated formulation of a load-plan. The agents then collaborate among themselves to place the cargo in such a manner that there are guaranteed to be no violations. Cargo items that could not be placed in any stow area without causing a violation are simply not loaded and identified to the user along with applicable details. Taking the vessel conveyance domain as an example, brief summaries of the functional capabilities of each ICODES agent are provided below.

The ***Stow Agent*** supports both manual and automatic load-planning operations. Using default settings in the automatic mode (i.e., Assisted Stow), the Stow Agent attempts to place the heaviest cargo items as low as possible on the ship without causing a violation. This results in a low center of gravity for the ship, which is desirable in most cases. The Assisted-Stow mode provides a comprehensive set of settings. This allows the user to define exclusive and inclusive constraints and preferences in respect to both the cargo that is required to be loaded and the stow areas that have been designated as being available. The Stow Agent checks to see that the placement of a cargo item does not overlap another cargo item, a fixture of the ship such as a stanchion or fire lane, or if the item is not entirely within a stow area. In Assisted-Stow mode, the user can also set the front/back and side to side spacing requirements of a cargo item (e.g., 18 inches front and back and 6 inches side to side) and the Stow Agent will abide by these settings so as not to place items within that imagery buffer around each cargo item.

Other parameters checked by the Stow Agent include the ports of embarkation and debarkation to ensure that they match the ports indicated in the voyage documents, and the height of each cargo item to ensure that the latter can reach their final locations. The Stow Agent automatically adds a safety cushion (specified by the user) to the actual height, which is set by the end-user, to make sure that height plus the cushion does not exceed the maximum allowable height for cargo in that stow area and the access path to the stow area.

In the Assisted Stow mode ICODES ensures that the automatically generated load-plan has no violations. In the manual mode (i.e., User Stow), ICODES will allow the user to load cargo items that are in violation. However, the Stow Agent will alert the user of the violations and provide an explanation.

The ***Trim and Stability Agent*** checks the placement of cargo items on the conveyance to see if they violate any desired (i.e., user specified) or mandated parameters, such as maximum draft settings, strengths (i.e., bending of the ship) or deck stress limitations in the case of vessels. The Stow Agent in automatic mode will rearrange the placement of cargo during the Assisted Stow process if the placement of cargo causes the upper limits of the strengths properties of the ship to be exceeded. For example, if the predefined loading order requires the center two stow areas of a deck to be loaded first and second, this would result in a sagging condition of the deck. Under these conditions, the Stow Agent will automatically redefine the loading order used by the Assisted-Stow process, so that the placement sequence of the cargo will begin with the forward and aft areas of the deck (thereby preventing the occurrence of a sagging condition).

ICODES calculates the effects of the exact placement of every cargo item loaded on the ship in three different planes. These planes are: forward to aft often referred to as the Longitudinally Center of Gravity; side to side or Transverse Center of Gravity; and, up and

down or Vertical Center of Gravity. The Trim and Stability Agent takes into account the combined effects of all of the cargo items, the ballast, and the original condition of the ship to provide the user with fairly accurate estimates of the center of gravity in each of the three planes, as well as an overall assessment of the stability of the ship.

The **Access Agent** checks all paths to ensure that a cargo item can be moved into a particular stow area. This includes openings, doors and hatches, differentiating between cargo that is loaded with cranes through hatches (i.e., LOLO: Lift On Lift Off) and cargo that is driven or pulled into stow areas (i.e., RORO: Roll On Roll Off). If in the Assisted Stow mode and if there is a violation in the loading path of a particular cargo item, the Stow Agent will not place this cargo item in that stow area but will attempt to place it in another stow area. In this situation the violation is transmitted indirectly from the Access Agent to the Stow Agent without notification of the user.

In manual mode (i.e., User Stow), on the other hand, if a cargo item is placed in a particular stow area for which all of the possible loading paths register an access violation, then the Access Agents will inform the user that the cargo item has a violation for every path to the final location. In addition, the Stow Agent will identify for the user the shortest loading path and the nature of the violation that is associated with that path.

ICODES allows the user to edit the ship characteristics, including the usability properties of the cranes and the dimensions of doors, openings and hatches. Since the Access Agent utilizes the current ship characteristics as the existing constraint conditions, these changes will be reflected in the actions of the Stow Agent in automatic mode and the alerts provided by the Access Agent in manual mode.

The **Cargo Agent** checks the characteristics of each cargo item against the expected characteristics for that cargo item recorded in the Marine Equipment Characteristics File (MECF) or Tech Data cargo libraries. Not all cargo characteristics can be verified in this manner. These cargo libraries currently contain more than 20,000 items, but are restricted in terms of the attributes that are provided for each cargo item. Typically, this verification process is complete and reliable only for dimensional (i.e., length, width and height) and weight attributes. If discrepancies are detected the Cargo Agent generates warnings.

The **Hazard Agent** verifies the proper placement of hazardous cargo items in reference to the various hazardous material codes and regulations discussed previously. In the case of vessels, it considers issues such as: Is the cargo item in an acceptable deck location according to its loading requirements? What are the segregation requirements for the cargo item, taking into account both the type of cargo item (e.g., break-bulk, container, vehicle) and the proximity of any other hazardous cargo items? Again in the vessel domain, for containers the Hazard Agent considers the hazard category of each item in the container in assessing the hazard condition of the container and its location relative to any other hazardous cargo item on the ship.

To effectively perform the extensive analysis described above the ICODES agents depend on a rich information model capable of providing the necessary domain knowledge and ever-changing context as the load-plan evolves. This critical content is provided by a set of domain models, or ontologies, which sets ICODES apart as an information-based, decision-support system.

Representation of Context – ICODES Ontologies

As a cohesive set of domain models the ICODES ontology in its entirety encompasses the concepts and entities that essentially comprise the view that ICODES has of the *world of load-planning*. In other words, the ICODES domain models provide expressive, context-oriented descriptions of both the tangible entities (e.g., conveyances in terms of loading areas, on-board facilities, etc., and cargo in terms of geometry, weight, etc.), as well as the intangible concepts (e.g., hazardous constraints, mobility, preference, accessibility, sequencing, etc.), and the large number of relationships necessary to support the decision-support capabilities offered by the ICODES suite of tools.

The ICODES model contains a number of sub-domains, including the Vessel Domain, Air Domain, Rail Domain, Yard Domain, Cargo Domain, and Plan Domain which describes the logistics of space planning in general. The following sections discuss three of these domain models in more detail; namely, the Vessel Domain, Cargo Domain, and Plan Domain.

Vessel Domain

As the name implies, the ICODES Vessel Domain model (Figure 3) provides a logistically biased view of a vessel. This representation has evolved substantially since the first version of ICODES was released in 1997. Both the advances in maritime technology, as well as the increasing demands from the user-base to support different types of load operations and cargo types, have triggered the evolution of the Vessel Domain model. A clear example of this need for enrichment of the Vessel model is the increase use of containers in the world of maritime transportation over the past 10 years. This requirement has resulted in model extensions to support container cells and tiers, as well as different container cell numbering systems such as the Baplie and MILSTAMP conventions. The following is a description of each of the primary elements comprising the Vessel model:

Stow Areas and Zones: The stowable space in the vessel is represented by *StowAreas* that keep track of the weight and area occupied by the cargo located in them. Further, to facilitate trim and stability calculations, *StowAreas* also record their center of gravity. Existing as a less formalized loading area, *Zones* allow the user to represent a subsection of a *StowArea*. Among numerous other attributes, *Zones* record their maximum cargo height and maximum deck stress. They can be used to represent a variety of loading conditions, each imparting their distinct semantics. For example, *NoStowZones* can be used to demarcate areas of the vessel where cargo should not be placed. Likewise, *OffLoadZones* can be used to represent temporary loading locations external to the vessel where cargo is to be off-loaded.

Access Entities: The Vessel model supports many different types of access entities, as those parts of the vessel that have to be traversed by cargo items as they move to their final location on the vessel. Examples of such access entities include booms, bulkheads, doors, elevators, hatches, openings, and ramps. These transition points must be richly described within the vessel domain in order to allow analytical agents to both identify and resolve potential accessibility issues.

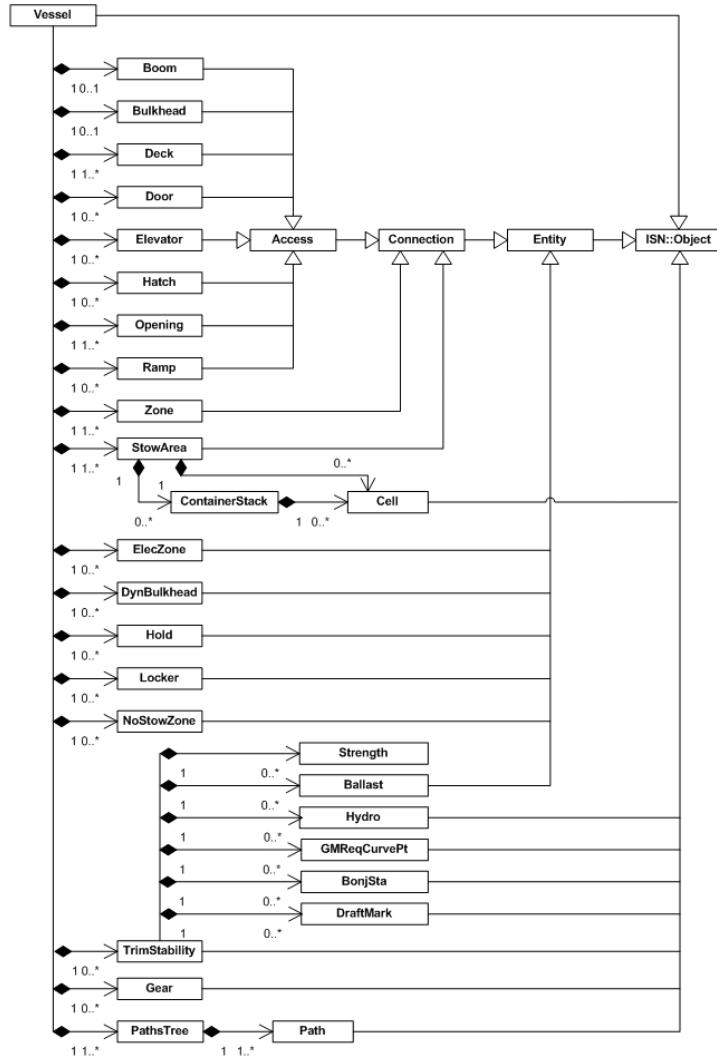


Figure 3: The Vessel Domain model

Trim and Stability: Most of the trim and stability information in the vessel domain is derived from a standard Ship Data file (i.e., SDA file) provided by the Maritime Administration (MARAD). The vessel model contains expressive descriptions of available tanks, ballast, hydrostatic properties, bonjean curves, strength, and draft marks critical for accurate trim and stability analysis.

Container Representation: The Vessel model employs the notion of *container tiers* to support vessels that are equipped to transport containers. Each container tier has a collection of container cells that represent the locations where individual containers can be placed. These container tiers provide the user with a top down view of the locations where container can potentially be placed. Container cells can also be grouped into *container stacks* and *container bays* that provide a cross-sectional representation of where containers can be placed. Supporting referential standards, these container cells can be identified by either Baplie or MILSTAMP numbers.

Ship Gear: The Vessel Domain also supports the notion of *gear*. As part of a particular vessel such equipment can be used in support of loading operations. Examples include forklifts, pallets, sweepers, and scissor trucks. As an integral part of the ICODES trim and stability analysis, the Vessel model representation of such on-board gear includes both weight and dimensional information.

Cargo Domain

The ICODES Cargo Domain model encompasses data and relationships directly related to the placement and tracking of cargo items (Figure 4). The representation of *cargo* within this domain includes not only the dimensional parameters of each cargo item, but also descriptions of hazardous material constraints (if any) and other logistical information required to effectively support the agent-based decision-support capabilities housed within the ICODES space-planning environment.

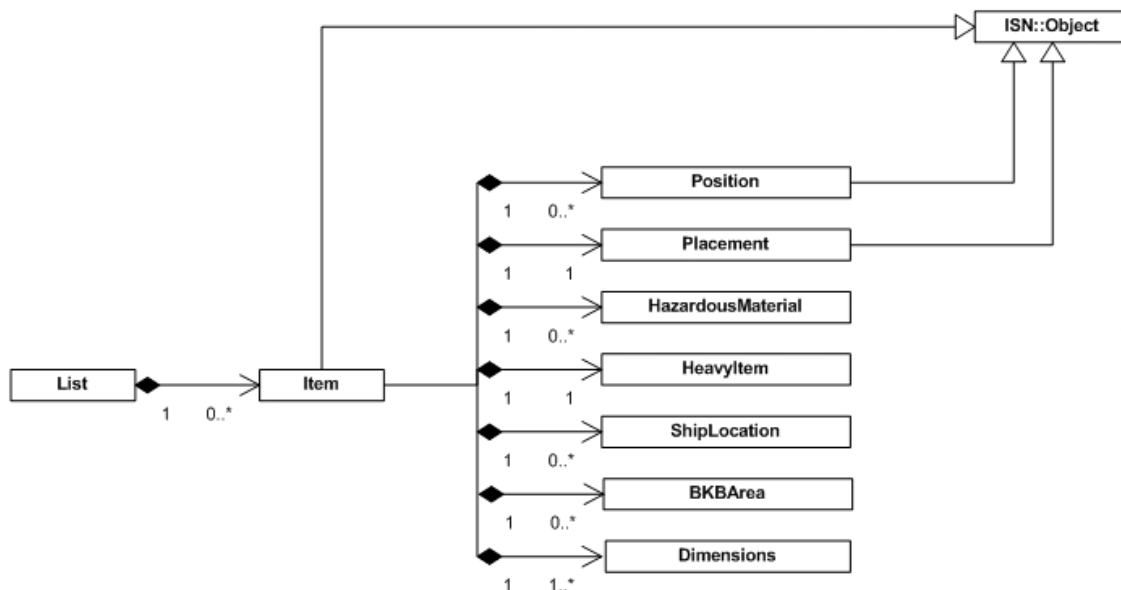


Figure 4: The Cargo Domain model

Each cargo item records information about its location to support the precise tracking of cargo as it moves from staging area to berth and through the ship to its final location. As such, items may contain multiple locations representing not only their current position but also where they have come from and where they are going and they are moved from staging area to their location on the ship. Within the user interface, ICODES uses ghost images of cargo to provide users with an easily discernable view of this dynamic locality information.

As mentioned earlier, cargo items include one or more hazardous information objects as integral parts of their description. Each of these objects represents information about the particular hazardous material(s) that comprise them.

Although ICODES allows the user to create a complete cargo list from inception, it is a more typical procedure for the cargo list to be imported from an external system such as the Transportation Coordinator's Automated Information for Movements System II (TC-AIMS II), the Marine Air Ground Task Force Deployment Support System II (MDSS II), or the Global Air

Transportation Execution System (GATES). Each of the external systems that ICODES supports may have several attributes unique to that system. Through various formal interface agreements, ICODES supports such attributes and is capable of displaying the information in the same fashion as ICODES' native attributes. Since these external attributes are considered *pass-through* data from one system to another, ICODES only provides a limited ability for the user to alter these values.

Plan Domain

The ICODES Plan Domain essentially aggregates all of the information relating to a load-plan into a cohesive Plan (Figure 5). Some of the information contained within an ICODES Plan includes relevant conveyances, cargo, and outstanding agent reports including violations. With the exception of agent reports, which are represented within this domain, each of these elements is described in terms of the Vessel Domain and Cargo Domain models discussed above.

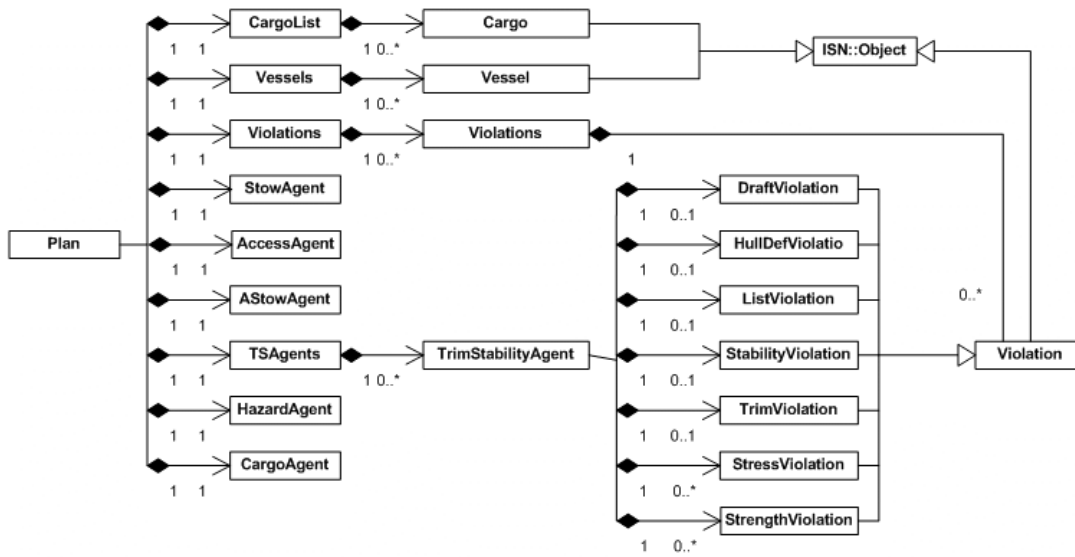


Figure 5: The Plan Domain model

Collectively, these domain models form the ICODES ontology. Each model represents a distinct set of information, concepts, and binding relationships relevant to specific aspects of load-planning operations. This ontology provides the context within which agents continuously analyze the evolving load-plan to identify emerging issues, explain those issues to users, and suggest mitigating actions.

Operational Performance Assessment

Empowered with a context-oriented representation that feeds intelligent software agents, ICODES provides its user communities with a technologically progressive approach to C2. It is generally accepted within the military load-planning community that ICODES has been responsible for a dramatic improvement in decreasing the loading time of ships and berthing costs. In addition, ICODES further proved its utility in unanticipated areas, such as ship selection for the movement of supplies, cargo in-transit visibility, historical analysis of cargo movements, and ship design. The following selected areas of military load-planning operations may serve as

indicators of the improvements in operating efficiency and cost savings that have been achieved through the deployment of the ICODES suite of adaptive tools over the past several years.

Load-planning efficiency: Previous to the fielding of ICODES in 1997, the creation of a load-plan would often take one load-planner using the DOS-based Computerized Deployment System (CODES) software at least two days. Once the cargo list had been cleansed, through the laborious manual process of comparing the data pertaining to each cargo item with the official equipment library, often a multi-day process, the load-planner would copy-and-paste the cargo symbols on the ship deck drawings. Then other planners with expertise in hazardous cargo, trim and stability, and cargo flow would check the plan, which often took another day. This time consuming cycle would begin again for each time the cargo list was updated, often up to 30 times during the development of a load-plan.

With ICODES, and in particular through its agents (i.e., Cargo, Access, Trim and Stability, Hazard, and Stow Agents), a load-planner is able to create a similar load-plan in about three hours. When updated cargo lists arrive, the ICODES *merge* function allows the same plan to be updated within minutes without re-starting the planning process.

Marine Corps cargo specialists have indicated that prior to the availability of ICODES the planning of the equipment for a force involving 10 to 14 ships would take an Operation Planning Team five to seven days. With ICODES this task has been reduced to about 14 hours.

In-transit visibility: An area of support that did not exist prior to ICODES is the electronic submission of cargo manifests and cargo ship placement reports to the ship personnel and the Port of Debarkation (POD) staff. This capability has provided visibility of cargo on the ship to assist with in-transit issues, to the POD for off-load planning and/or load-planning of new loads, and to military administrative personnel for tracking and historically reporting on cargo movements.

At a POD, prior to ICODES, immediately after the arrival of a vessel, a cargo survey and meeting would be held to discuss cargo placement and off-loading strategies. With the availability of ICODES documentation, this half-day delay is no longer necessary resulting in a significant saving of berthing costs. In addition, the off-load planning that can now be accomplished with ICODES prior to ship arrival results in substantial labor and off-load space assignment savings.

For ships with multiple ports of loading and discharge, ICODES load-plans are now passed electronically from port to port so the effects of the loads and off-loads on the ship can be determined and personnel in different ports can have a common operating picture. Beyond the port, the Army Logistic Operations Center uses a database of ICODES-generated load-plans to estimate off-load times. In the past this has been a labor intensive operation, often resulting in missed deadlines.

Trim and stability analysis: Since the ICODES Trim and Stability Agent utilizes certified formulas for ship trim and stability calculations, the results are not only used by load-planners but also by the ship's crew to confirm ship loading conditions. Because of the trusted quality of the validated ICODES trim and stability analysis, ships are much less prone to unsafe load configurations and further, sail up to a day earlier than in the pre-ICODES era. The earlier departure of ships leads to fuel savings since ships are able to proceed at reduced

speed and still stay on schedule. In addition, ships loaded with the precision and operational knowledge offered by the ICODES system experience decreased port costs associated with berthing and service fees.

Prior to the availability of ICODES, ships were often loaded with little concern for the distribution of weight along the ship's perpendicular axis, eventually causing several classes of ships to develop stress fractures. The continuous monitoring of the condition of the ship during load-planning has led to better load distributions and the resultant reduction in costly ship repairs.

Reconciliation of planned cargo placement: Using the ICODES Automatic Information Technology (AIT) capabilities, the staging area cargo placement and the ship as-loaded plan is confirmed by people with hand-held Personal Digital Assistants (PDA), as opposed to people using manually drawn sketches and tally sheets. Using the ICODES AIT functionality, personnel costs have been reduced about 80%, i.e., to about 20% of the cost of the manual process, and the number of port cargo administrative personnel have been reduced by about 50%. With the increasing availability of AIT wireless communications at ports cargo, locations are updated automatically to an ICODES computer in the port command center, allowing near real-time visibility of cargo to port administrative personnel and preventing the misplacement of hazardous materials.

Since its first release as a system of record in 1997, the granularity of the cargo data has increased greatly as ICODES moved from Level 4 to Level 6 detail. A typical Army cargo list in 1997 seldom included more than 2,000 individual cargo items. From 2004 onward ICODES has been required to process Marine Corps cargo lists with more than 30,000 individual cargo items. Despite this increase in the volume of data the performance of ICODES, in terms of response time, has continued to improve. The typical performance results shown in Table 1 are based on periodic metrics collected by the ICODES Program Management Office over the past eight years.

Table 1: Historical ICODES performance metrics

Tested Procedure	V 3.0 (1998)	V 5.0(2001)	V 5.4 (2005)
Create two-ship load-plan with 2,400 normal cargo items	20 min	8 min	1.5 min
Create two-ship load-plan with 1,200 hazardous cargo items	25 min	11 min	2.5 min
Unload inventory of 2,400 items from two ships	10 min	5 min	1.0 min

In-Field Impressions

With a user-base of over 2,500 military users that spans across the entire globe, ICODES is an example of the successful employment of an ontological approach to logistics and C2. This success highlights the capabilities and extensive decision-support that the use of ontologies can provide to C2. Significantly increasing the efficiency and accuracy of logistics and C2 operations, ICODES has been received exceedingly well by its relatively large user-base as an effective decision-support tool for load-planning. For example, a Marine Corps Captain wrote after using ICODES:

"My battalion [...] used ICODES to support the embarkation for our [...] deployment in 2005 with the [...] MEU [Marine Expeditionary Unit]. This deployment included dozens of onloads and offloads with over 200 pieces of rolling stock. We also used the program to support disembarking in preparation for movement into Iraq and our subsequent retrograde. We were one of the first Marine Corps units to use ICODES on deployment and it reduced the time required to develop a ship load plan and greatly simplified the onload process. It also made it easy for the U.S Navy to quickly understand how we wanted our equipment loaded on the following amphibious warships and landing crafts: LHA, LHD, LSD, LPD, LCACs and LCUs.

"More importantly, once the ships were loaded, the battalion had a very accurate computerized plan of a how its equipment was tactically loaded. If my battalion commander received an unexpected mission, he would call on our Embarkation Officer to give him an estimate of the tactical offload process. These estimates were developed using ICODES and proved very accurate in planning for a tactical amphibious assault or raid. ICODES truly proved its reliability and viability with us on this deployment."

Providing information to commanders in support of their decision making is the essence of C2. ICODES has been used by the Marines for C2 as well as for the technical work of planning ship loadouts as exemplified by the following quotation from another Marine Corps Captain:

"As far as how we used ICODES, in the embarkation world it was a necessity. We used it constantly during our deployment on the [...] MEU [...] for the planning & execution of onloads, offloads, and the re-arrangement of vehicles & cargo aboard ship while we were underway.

"The version we used in 2004-2005 was fairly accurate in terms of reflecting the actual dimensions of the cargo we loaded, which was the critical part as we utilized every inch of space we had. ICODES not only provided us (the embark people) with a good planning tool, but also painted an easy-to-understand picture for higher ups and other staff and commanders that would be helping execute the plan."

This usage included relief operations after the 2005 Tsunami in Indonesia.

"We not only used ICODES during the tsunami relief, but in some ways use of the program was more critical during this period than during some of our more basic administrative or tactical offloads. Even though we didn't send a lot of our gear ashore during the tsunami relief (more food & water than gear), we nevertheless had to re-arrange our embarked vehicles and cargo to allow for enough room to load supplies onto the landing craft.

"Although the commanders never used ICODES, we often showed them Powerpoint briefs copied from ICODES so they could be aware of how the ships were currently loaded & what gear would be available for offload first."

ICODES has proven to significantly shorten the OODA loop (i.e., Observe, Orient, Decide, Act) by increasing in-transit visibility, automatically generating warnings and alerts with associated explanation, providing intelligent decision-assistance tools for the development and evaluation of plans, and most importantly providing commanders with easy-to-understand images.

The Next Generation ICODES – ICODES v6

A critical requirement for the ICODES suite of load-planning services is the ability to grow to meet increasing needs. With an initial narrow focus ICODES was designated as the United States (US) Department of Defense *migration system* for ship load-planning in 1996. However, as the user-base of ICODES increased so did the number of requests to support specialized problems and application domains that were not considered in the original design of the ICODES toolset. In November 2007, after an extensive evaluation of alternatives, ICODES was designated by the US Transportation Command's (USTRANSCOM) Distribution Steering Group (DSG) to become the Single Load-planning Capability (SLPC) for all types of conveyances. Consequently, by 2011 ICODES v6 Global Services (GS) is expected to provide planning and execution support for cargo movement by ship, rail, air, trucks, warehousing, staging, and other domains that require space planning and in-transit visibility capabilities (Figure 6). The foundations of each of these additional domains exist as carefully crafted ontology domains that express the concepts and tangible features which allow software agents to effectively reason about the particular load-planning activity at hand.

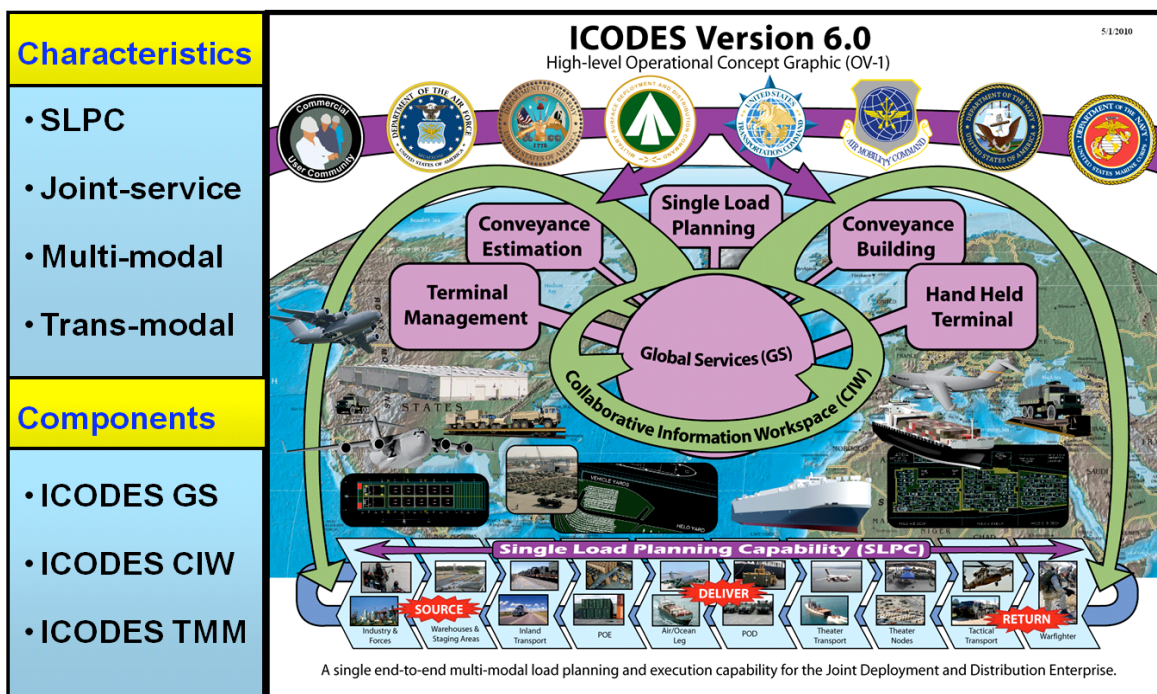


Figure 6: ICODES v6 operational vision

Although designed to support an expanded set of functionality necessary to accommodate this multitude of transportation modes, ICODES v6 must also be architecturally ready to integrate additional capabilities or services, such as viewers tailored to specific operational needs, critical data feeds from external sources, and newly available capabilities such as *smart tags* and other emerging technologies. Designed as a Service-Oriented Architecture (SOA) solution allows ICODES GS to incorporate such additions in a manner that is efficient and preserving of overall system integrity. Figure 7 illustrates the collection of layered services comprising ICODES GS.

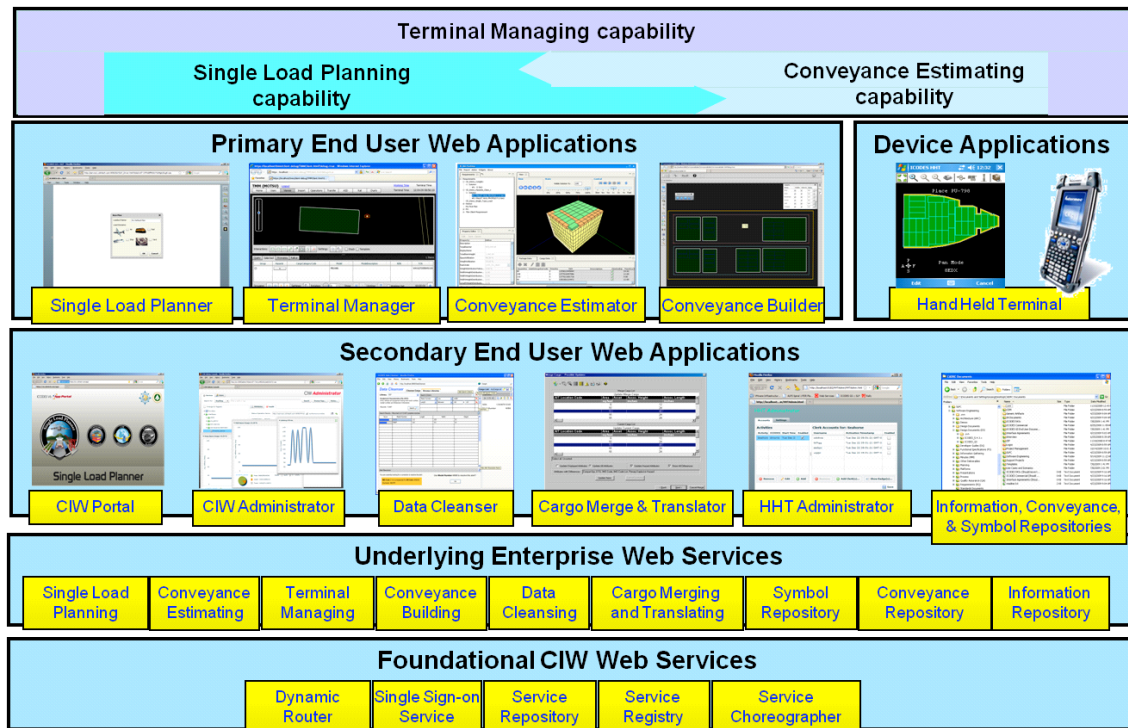


Figure 7: ICODES v6 as a collection of layered services

ICODES GS is designed to be deployed in three distinct forms. The first of these deployment models is to SDDC's Common Computing Environment (CCE) enterprise. Within this deployment, ICODES GS services will be hosted and managed alongside other enterprise services within a virtualized computing environment. As this environment evolves into a fully capable cloud computing environment, the SOA design of ICODES GS will allow such capabilities to seamlessly transition to this distributed enterprise.

The second deployment model supported by ICODES GS is similar in form to the CCE except that it is targeted to installations that require a higher degree of control over their computing environments, such as is often the case with terminal management operations. These *satellite* installations still benefit from local distributed computing environments and make particular use of the Terminal Management Module (TMM) capability available within ICODES GS.

The third deployment model supported by ICODES GS is targeted directly to those operating environments that exhibit limited ship-to-shore connectivity, such as is often the case for operations at sea.

However, with simplicity and manageability in mind, all three of these deployment models use the exact same ICODES GS software code. Tailoring this suite of capabilities to the particular computing environment is managed externally by the installer. Adopting this approach allows the application code comprising ICODES GS to be essentially agnostic as to the particular form of the deployment. This approach avoids the complexities involved in maintaining specific application code dedicated to particular deployment models.

The Challenges of Effectively Supporting C2

The operational and economic benefits of ICODES suggests that it is a matter of when, not if, DoD will develop and implement a C2 ontology. However, there are four significant aspects of C2, Congressional guidance, and DoD guidance that must be understood by those developing a C2 ontology.

First, there are authoritative definitions of C2 as well as associated systems through which C2 is implemented. The definition of C2 is given at the beginning of this paper. The authoritative definition of a C2 system according to Joint Publication 6-0, *Joint Communications System*, is:

“The facilities, equipment, communications, procedures, and personnel essential to a commander for planning, directing, and controlling operations of assigned and attached forces pursuant to the missions assigned.”

Those developing a C2 ontology must *accept, understand, and work within* these definitions. Although it may be tempting to “improve” upon them, basing a C2 ontology on a set of definitions that differ from those adopted by warfighters threatens to reduce the acceptance and effective utility of the resulting model.

Second, Congress and senior DoD leaders are providing very specific guidance on the execution of IT and C2 efforts. These include a requirement for those developing a C2 ontology or other elements of information technology systems to (a) support rapid prototyping of new or improved capabilities and (b) gain and maintain a close relationship with users to insure delivery of products the users find effective. In response to the latter of these mandates, efforts to build C2 ontologies will likely need to explicitly demonstrate how development is achieving a strong connection with users. Congress’s guidance on this concern is outlined in the Fiscal Year 2010 National Defense Authorization Act (NDAA). Section 804 of that act directs:

“The Secretary of Defense shall develop and implement a new acquisition system for information technology systems. The acquisition process developed and implemented pursuant to this subsection shall, to the extent determined appropriate by the Secretary—

be designed to include—

- (A) early and continual involvement of the user;*
- (B) multiple, rapidly executed increments or releases of capability;*
- (C) early, successive prototyping to support an evolutionary approach; and*
- (D) modular, open-systems approach.”*

In addition, DoD guidance regarding this issue is found in the “DoD Command and Control Strategic Plan” which states:

“While advanced concepts and technologies associated with net-centricity can enable seemingly ubiquitous access to information and unprecedented situational awareness and timely collaboration among mission partners, the Department’s efforts in the C2 capability area will still be guided by the principal maxim of command and control: that

technology enables the human interface and supports “command” and the decision-maker, rather than forcing the decision-maker to operate within the constraints of the “control” technology. The force development community will remain cognizant of this to ensure C2 technical solutions meet commanders’ needs.”

This critical direction is further reflected in the “DoD Command and Control (C2) Implementation Plan” which states:

“The Department’s efforts in the C2 capability area will be guided by the principal maxim of command and control: technology enables the human interface and supports ‘command’ and the decision-maker, rather than forcing the decision-maker to operate within the constraints of the ‘control’ technology. The force development community will remain cognizant of this to ensure C2 technical solutions meet commanders’ needs.”

It is further likely that the new acquisition system for information technology systems that the FY 2010 NDAA requires the Secretary of Defense to develop will include a requirement for developers of a C2 ontology to be able to explain how they are keeping warfighters involved. In large measure, ICODES has been successful because from its very inception in the mid-1990s, the ICODES development team embraced the same user-centric philosophy that is now being formally reflected in such Congressional and DoD guidance.

Third, warfare is a non-deterministic activity, a battle of wits. Although the equipment of warfare attracts much attention, success is achieved by the side whose commanders outwit the commanders of the opposition. If we are proficient at performing a certain activity or perhaps well-equipped to process a particular category of data into actionable information, our adversaries will seek to neutralize such abilities by shifting the conflict towards activities where such skills are essentially irrelevant, or perhaps constantly require adaptation. The agility necessary to stay one-step-ahead of the enemy inevitably involves equipping commanders with tools capable of providing meaningful information and intelligent analysis capabilities that can quickly adapt to the dynamics of warfare. While success within this arena is ultimately a matter of commanders outwitting those of the opposition, technology can play a vital role in equipping commanders with an arsenal of capabilities to support their decision-making.

Fourth, C2 systems must seek to free commanders to outwit the enemy. Warfare is inherently a complex, data-intensive activity. As a result, data-overload can be a common scenario facing decision-makers. Ontology-based C2 tools can help commanders make sense out of this sea of data by providing the means to quickly build a concise operational picture. In this manner, users are not only freed from much of the routine processing involved in achieving such rapid and focused situation awareness, but are further equipped with analysis capabilities (e.g., communities of intelligent agents) that can significantly help offset the effects of an increasingly reduced DoD staff.

Conclusion

The ICODES application currently provides a comprehensive tool-set of software agents to assist the cargo specialist in the development of ship load-plans for military deployments. It is one of the earliest military examples of information-centric software that incorporates an internal,

relationship-rich information model to provide context for the reasoning functions of collaborative software agents. With the upcoming release of ICODES GS, functionality along with the ICODES ontology will be extended to support the load-planning of all types of conveyances (i.e., ships, aircraft, trains, and trucks) and assembly areas. At the same time the ability of its underlying SOA-based design will be severely tested as ICODES scales from a stand-alone application to a global environment of integrated intelligent services.

As an information-oriented, agent-based system, ICODES adheres to three notions that are fundamental to its decision-support capabilities.

1. ICODES processes information (i.e., data with relationships) as opposed to legacy systems that normally process data only (even though the data may be in the form of objects with characteristics). The key to the assistance capabilities of ICODES is that the system has some understanding of the information that it is processing. In the ICODES ontology cargo items are described in terms of characteristics that relate each item to hazard, trim and stability, accessibility, and ship configuration, constraints. This internal information model provides context for the automatic reasoning capabilities of software agents.
2. ICODES is a collection of powerful collaborative tools, not a library of predefined solutions. This overcomes the deficiencies of legacy systems in which built-in solutions to predetermined problems often differ significantly from the complex operational situations encountered in the real world. In this respect ICODES is a collaborative decision-support system in which the operator interacts with computer-based agents (i.e., decision making tools) to solve problems that cannot be precisely or easily predetermined.
3. ICODES incorporates agents that are able to reason about the characteristics and the relationships of cargo items, the internal configurations of conveyances and the constraints that must be considered during the development of load-plans. Although these agents are decoupled (i.e., do not know about each other's existence) they are able to indirectly collaborate through a data blackboard and subscription services, as they assist the user throughout the load-planning process.

The advantages of an information-centric software system have been evidenced in three areas by the performance of ICODES in the field over the past several years. First, if all necessary data are available, ICODES is capable of automatically generating the load-plans of four medium-size ships in around two hours. This is a significant improvement in load-planning speed over the legacy application that it replaced. The predecessor application typically required two person-days for the development of a single load-plan. Second, the assistance capabilities of the ICODES agents elevate the performance of a novice load-planner to at least an acceptable level. This is an important consideration in view of the attrition rate of military cargo specialists during the past decade. The performance of an expert load-planner, on the other hand, is raised to an exceptionally high productivity level. And third, the ability of ICODES to continuously evaluate the evolving load-plan in respect to accessibility, hazardous material, and trim and stability conditions, has greatly increased the quality and accuracy of the resulting load-plan.

References

- Bradshaw J. M. (ed) (1997); 'Software Agents'; AAAI Press / The MIT Press, Massachusetts, (pp. 3-11).
- Diaz C., W. Waiters, J. Pickard, J. Naylor, S. Gollery, P. McGraw, M. Huffman, J. Fanshier, M. Parrott, S. O'Driscoll-Packer, Boone Pendergrast and Evan Sylvester (2006); 'ICODES: Technical and Operational Description'; Technical Report CDM-20-06, CDM Technologies Inc., San Luis Obispo, California, November.
- Fowler M and K Scott (1997); 'UML Distilled: Applying the Standard Object Modeling Language'; Addison-Wesley, Reading, Massachusetts.
- Mowbray T. and R. Zahavi (1995); 'The Essential CORBA: Systems Integration Using Distributed Objects'; Wiley, New York, New York.
- MTMC/ICODES (2002); 'ICODES Version 5.2: USMC Basic Training Manual' and 'ICODES Version 5.2: Advanced Training Manual'; Military Traffic Management Command (MTMC), US Army.
- Myers M. and J. Pohl (1994); 'ICDM: Integrated Cooperative Decision Making – in Practice'; 6th IEEE International Conference on Tools with Artificial Intelligence, New Orleans, Louisiana, November 6-9.
- Pohl J., A. Chapman, K. Pohl, J. Primrose and A. Wozniak (1992); 'Decision-Support Systems: Notions, Prototypes, and In-Use Applications'; Technical Report CADRU-11-97, Collaborative Agent Design Research Center, Design and Construction Institute, College of Architecture and Environmental Design, Cal Poly, San Luis Obispo, California, January.
- Pohl K. (2001); 'Perspective Filters as a Means for Interoperability Among Information-Centric Decision-Support Systems'; Office of Naval Research (ONR) Workshop hosted by the Collaborative Agent Design Research Center (CADRC), Cal Poly (San Luis Obispo, CA) in Quantico, Virginia, June 5-7.
- Pohl J., K. Pohl, R. Leighton, M. Zang, S. Gollery and M. Porczak (2004); 'The ICDM Development Toolkit: Purpose and Overview'; Technical Report CDM-16-04, CDM Technologies, Inc., San Luis Obispo, California, May.
- Pohl J. (2007); 'Knowledge Management Enterprise Services (KMES): Concepts and Implementation Principles'; Pre-Conference Proceedings, Focus Symposium on Representation of Context in Software, InterSymp-2007, Baden-Baden, Germany, 31 July.
- Pohl K. (2002); 'The Underlying Design Principles of the ICDM Development Toolkit'; Preconference Proceedings, InterSymp-2002, Focus Symposium on Collaborative Decision-Support Systems, Baden-Baden, Germany, July 29-August 30.
- Shreiner D. (ed.) (2000); 'Open GL Reference Manual'; Addison-Wesley, Menlo Park, California.
- Wernecke J. (1994); 'The Inventor Mentor: Programming Object-Oriented 3D Graphics with Open Inventor'; Addison-Wesley, Menlo Park, California.
- Wooldridge M. and N. Jennings (1995); 'Intelligent Agents: Theory and Practice'; The Knowledge Engineering Review, 10(2), (pp. 115-152).

An Intelligent Supply Chain Planning and Execution Environment

Jens Pohl, PhD

Executive Director, Collaborative Agent Design Research Center (CADRC)

Director, KML Center (Scott AFB, Illinois)

California Polytechnic State University (Cal Poly), San Luis Obispo, California

Abstract

Logistic planning and execution processes in a supply-chain are subject to a high level of complexity because of the number of parties and issues involved, the number of relationships that exist among them, and the dynamic nature of the execution environment. The large volume of data flowing through a sizable computer-based logistic planning and execution management environment that is based on rote data-processing principles tends to overwhelm the human users. As a result many opportunities for improving the efficiency of supply-chain processes and thereby reducing costs are overlooked by the human users, who are forced into a reactive mode.

Similar data deluge symptoms are being experienced in other domains such as Internet searches where the number of website hits returned for a single query can easily exceed several million. The data deluge problem could be overcome if the context of the query could be defined by the user and executed by the search engine in a context-based manner. This would require the representation of a virtual model of real world context in the search software. The same need for the representation of context in software exists also in the cyber security domain where data encryption must be supplemented by the profiling of users and the continuous monitoring and automated interpretation of network behavior.

This paper discusses the design concepts and implementation principles, and describes the end-state capabilities of a computer-based intelligent logistic planning and execution environment that includes a virtual model of real world supply-chain context and multiple agent groups that are able to interact with each other and the human users. Implemented in a service-oriented architecture (SOA) based infrastructure, the virtual context model provided by a multi-layer ontology and the collaborative agents are able to continuously monitor the state of the supply-chain by interpreting the flow of data in the appropriate context. This allows the agents to rapidly re-plan in case of supply-chain interruptions, discover and act on opportunities for improvements, and identify patterns and trends based on the continuous analysis of historical data. As a result the human users are relieved from lower level data interpretation tasks and provided with actionable information for reactive and proactive planning and execution management functions. The author suggests that order of magnitude improvements in efficiency and reduction in cost are achievable with context-based information-centric software systems.

1. Supply-Chain and Logistics

Organizations exist for some purpose and in virtually all cases this purpose involves the creation and delivery of products, in the form of goods and/or services. To achieve its purpose the organization uses a variety of resources such as people, information, materials and/or components, and money, to perform operations that result in the delivery of products to its customers. The required operations may include any number of activities such as manufacture, transportation, training, serving, and selling, and typically involve many activities and

relationships that need to be coordinated within a network of interacting entities. The Chartered Institute of Logistics and Transport (1998) defines supply-chain and logistics as follows:

“The supply-chain is a sequence of events intended to satisfy a customer. It can include procurement, manufacture, distribution and waste disposal, together with associated transport, storage and information technology.”

“Logistics is the time related positioning of resources or the strategic management of the total supply-chain.”

The principal objectives of supply-chain management are normally focused on optimizing the sequence of operations in combination with the resources that are required to perform the operations so that the expectations of the customer are satisfied at least cost to the organization. There are many factors that can make it difficult to achieve an optimum supply-chain management outcome (Waters 2007). The logistical functions involved comprise a series of related activities, including acquisition, receiving, warehousing, inventory management, order processing, transportation, distribution, and so on. The workflow processes involved are often quite complex and typically involve several parties with different skill sets and objectives. In a global supply-chain the need to move goods and services across national borders increases the potential complexity by an order of magnitude. At the same time the desire to minimize inventory increases the risk factor and makes it incumbent on the organization to proactively anticipate disruptive events and effectively react to disruptions when they inevitable occur.

A large scale global supply-chain is a very complex undertaking that involves a high level of risk (Handfield 2008, Handfield et al. 2008, Manuj et al. 2007). Much of the risk is associated with factors that cannot be directly controlled by the organization. These include unavailability of essential resources or components, inclement weather conditions, traffic congestion, custom delays at national borders, breakdown of essential equipment, terrorism and criminal activities, and unforeseen surges in customer demand that can all lead to unexpected disruptions of the end-to-end supply-chain. In recent years with the increase in customer expectations, competition, and political volatility the anticipation and ability to react under time critical conditions to such disruptions has placed an emphasis on effective supply-chain *event* management.

Clearly, such a complex, dynamically changing and time critical undertaking requires sophisticated information management support and can benefit greatly from automated monitoring, planning, tracking, and intelligent decision-assistance services. This paper proposes an enterprise-wide intelligent information management environment based on currently available computer hardware and software technology that is capable of providing the required level of support. It is generally understood that current operational trends and advances in information technology are inevitably leading to the eventual realization of the proposed information management capabilities. However, the opportunity exists to accelerate this progress and reap the significant business benefits that will accrue to the organization that captures the leading share of the supply-chain management software market that has been projected at \$5.5 billion in 2011 (AMR 2007).

2. The Inherent Complexity of Logistical Planning and Execution

Logistical planning and execution within a supply-chain can have all of the characteristics that are commonly associated with the family of *complex problems*. These characteristics include: many entities and issues that are related to each other; large volume of data that needs to be categorized and analyzed to extract useful information; the reliability of some of the data may be

questionable; incomplete data in some areas requiring time critical decisions to be made with partial information; and, a dynamically changing and largely unpredictable execution environment (Pohl 2008, 49-59).

Swaminathan et al. (1998) have identified two categories of supply-chain elements, namely structural elements and control elements. Structural elements such as vendors, manufacturers, suppliers, distribution centers, and conveyances are concerned with the acquisition, transportation and delivery of goods and services. Control elements such as demand and supply, inventory, routing, and the availability of information govern the flow of processes within the supply-chain. The interrelationships among these two groups of elements are responsible for the complex nature of the supply-chain. The degree to which these complex interactions can be effectively managed is greatly dependent on the accuracy of demand forecasting, the continuous flow of timely and reliable information, the availability of resources such as supplies and conveyances, and a host of external factors such as weather conditions, route closures, accidents, and criminal actions. These external factors are largely unpredictable and have the potential of severely disrupting the supply-chain, despite the most careful attention to planning and execution monitoring.

3. Desirable Capabilities of an Intelligent Supply-Chain Environment

Some importance is attached to the term *environment* in preference to the more conventional nomenclature that would refer to a related set of software components that are intended to interoperate as a *system*. The use of the term environment is intended to convey a level of integration of capabilities that is seamless and transparent to the user. In other words, persons engaged in the logistic planning, monitoring and decision-making processes should not be conscious of the underlying software and inter-process communication infrastructure that is necessary to support the operation of the environment. The objective is for the human users to be immersed in their management activities to the extent that both the automated capabilities operating mostly in background and the capabilities explicitly requested by the user at any particular time operating in foreground are an integral part of the process. Ideally, the human user should perceive the logistic management activities and the environment within which these activities are being performed as being synonymous.

From a general point of view there are at least two overriding requirements for an intelligent computer-based decision-making environment. The first requirement relates to the representation of information within the environment. The software must have some level of *understanding* of the information context that underlies the interactions of the human user with the environment. This is fundamental to any meaningful human-computer interaction that is akin to a partnership. The level to which this *understanding* can be elevated will largely determine the assistance capabilities and essentially the value of the software environment to the human user.

The second requirement is related to the need for collaboration. In a broad sense this includes not only the ability to interact with human stakeholders who play a role in the supply-chain, such as planning and management personnel, vendors, remote distribution centers, shippers, and customs officials, but also non-human sources of information and capabilities. All of these interactions between human participants in the logistic processes, data sources, and software-based problem solving capabilities, must be able to be performed seamlessly without the user having to be concerned about access protocols, data formats, or system interoperability issues.

While these overall requirements would at first sight appear to be utopian compared with the state of computer-based environments that exist today (2010), the technology needed for the creation of such environments has been rapidly emerging during the past decade and is now largely available. However, before addressing the technical software design aspects it is perhaps appropriate to delve more deeply into the functional requirements of an intelligent logistic planning and execution environment.

3.1 Emphasis on partnership

A desirable logistic information management environment is one that assists and extends the capabilities of the human user rather than replaces the human element. Human beings and computers are complementary in many respects. The strengths of human decision makers in the areas of conceptualization, intuition, and creativity are the weaknesses of the computer. Conversely, the strengths of the computer in computation speed, parallelism, accuracy, and the persistent storage of almost unlimited detailed data are human weaknesses. It therefore makes a great deal of sense to view a computer-based supply-chain environment as a partnership between human and computer-based resources and capabilities.

This is not intended to suggest that the ability to automate functional sequences in the computer-based environment should be strictly confined to operations that are performed in response to user actions and requests. Apart from the monitoring of problem solving activities, the detection of conflicts, and the execution of evaluation, search and planning sequences, the computer-based environment should be able to undertake proactive tasks. The latter should include not only anticipation of the likely near-term need for external data sources that need to be acquired by the environment, but also the exploration of alternative solution strategies that the environment considers promising even though the user may be currently pursuing another path.

In this partnership a high level of interaction between the human user and the computer-based environment is a necessary feature. It provides opportunities for the planning and management personnel to guide the environment in those areas of the decision-making process, such as conceptualization and intuition, where the skills of the user are likely to be far superior to those of the computer. Particularly prominent among these areas are conflict resolution and risk assessment. While it would be of considerable assistance to the human users to be alerted to conflicts and for the nature of the conflicts to be clearly identified, there are advantages for the resolution of such conflicts to be undertaken in collaboration with the users.

It follows that the capabilities of the computer-based environment should be designed with the objective of assisting and complementing the user in a teaming role. Such tools are interactive by nature, capable of engaging in collaboration with the user to acquire additional information to help better understand the situation being analyzed. These tools are also able to provide insight into the reasoning processes that they are applying, thereby allowing the human planners and decision-makers to gain confidence in their inferencing capabilities as well as make subtle adjustments in the logic being applied. The author's past experience with multi-agent decision-support applications has shown that tools that are engineered for collaboration with each other and the human user provide opportunities for augmenting their capabilities through user interaction during execution (Pohl et al. 1997). It is therefore suggested that these kinds of tools better assist the human users in dealing with the complexities of the logistic processes involved in the supply-chain. In other words, a collaborative approach affords the necessary visibility and agility to deal with the large number of considerations across a far reaching set of domains that characterizes the supply-chain.

3.2 Collaborative and distributed

Supply-chains, or complex problem environments in general, normally involve many parties that collaborate from widely distributed geographical locations and utilize information resources that are equally dispersed. A computer-based logistic planning and execution environment can take advantage of the distributed participation by itself assuming a distributed architecture. Such an architecture typically consists of several components that can execute on more than one computer. Both the information flow among these components and the computing power required to support the system as a whole can be decentralized. This greatly reduces the potential for communication bottlenecks and increases the computation speed through parallelism.

Another advantage of the distributed approach is the ability to modify some components of the system while the system as a whole continues to operate with the remaining components. Similarly, the malfunction or complete failure of one component does not necessarily jeopardize the entire system. This is not so much a matter of redundancy, although the distributed architecture lends itself to the provision of a high degree of redundancy, but rather a direct result of the physical independence of the components. While the components may be closely integrated from a logical point of view they can operate in their own autonomous physical environment.

3.3 An open architecture

The high degree of uncertainty that pervades complex problem environments, such as logistic planning and execution, extends beyond the decision-making activity of the collaborating planners and decision-makers to the configuration of the computer-based environment itself. The components of a design environment are likely to change over time, through modification, replacement, deletion, and extension. It should be possible to implement these changes in a seamless fashion through common application programming interfaces and shared resources. Service-Oriented Architecture (SOA) concepts align well with this principle by treating the required planning, monitoring, and decision-assistance functionality as a composition of discrete, self-contained software services with a very low degree of coupling between components (Erl 2008).

3.4 Tools rather than solutions

The computer-based logistics environment should offer a set of tools rather than solutions to a predetermined set of problems. The indeterminate nature of the supply-chain does not allow us to predict, with any degree of certainty, either the specific circumstances of a future problem situation or the precise terms of the solution. Under these circumstances it is far more constructive to provide tools that will extend the capabilities of the human decision-maker in a highly interactive problem solving environment.

In this sense a tool is defined more broadly than a sequence of algorithms, heuristics or procedures that are applied largely on the direction of a user. Tools can be self-activating, be capable of at least semi-autonomous behavior, and cooperate with each other and users in employing and providing services.

3.5 Expressive internal representation

The ability of the computer-based environment to convey a sense of having some level of understanding of the meaning of the data and in particular the concepts being processed is the single most important prerequisite for a collaborative information management environment

(Assal et al. 2009). An expressive representation of the real world supply-chain entities and concepts that define the problem space forms the basis of the interactions between the users and the information management environment and, also, the degree of intelligence that can be embedded within its components (Figures 1 and 2).

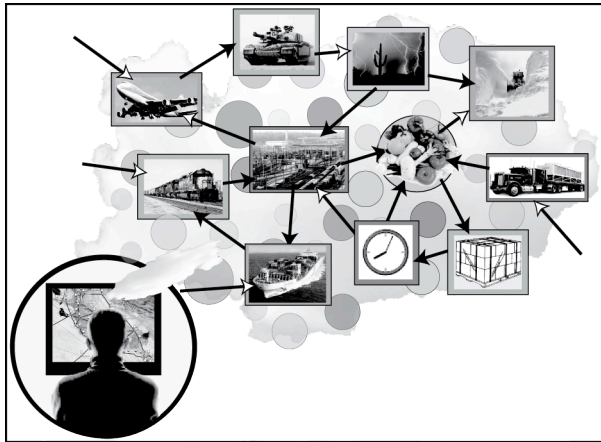


Figure 1: Virtual model of the supply-chain entities and their interrelationships

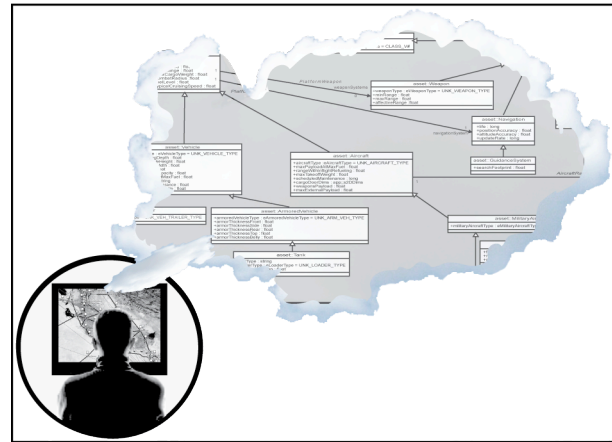


Figure 2: Ontology representation of the supply-chain that is machine processable

To the logistic planning and management personnel the supply-chain consists of real world entities such as requisitions, contracts, goods, services, conveyances, routes, points of embarkation and debarkation, distribution centers, schedules, delivery windows, and costs, as well as related concepts such as efficiency, security, performance, risk, and trust. Each of these notions has properties and relationships that determine their behavior under certain conditions. These semantic descriptors form the basis of collaboration among human problem solvers and are therefore likewise the fundamental subject matter of concern in an enterprise-wide collaborative logistic planning and execution environment.

3.6 Embedded knowledge

The computer-based logistic planning and execution environment should be a knowledge-based environment. In this context knowledge can be described as experience derived from observation and interpretation of past events or phenomena, and the application of methods to past situations. Knowledge-bases capture this experience in the form of rules, case studies, standard practices, and typical descriptions of objects and object systems that can serve as prototypes. Problem solvers typically manipulate these prototypes or patterns through adaptation, refinement, mutation, analogy, and combination, as they apply them to the solution of current problems (Gero et al. 1988, Pohl 2008).

3.7 Decentralized decision-making

While a global supply-chain can be centrally coordinated, the planning and management processes that are required for its efficient operation cannot be centrally controlled. Many of these planning and execution activities will be localized and performed in parallel involving the collaboration of different members of the supply-chain team. In this regard, due to its continuously changing nature, logistic execution is neither a rigidly controlled nor a strongly disciplined activity but rather a process of information seeking, analysis, collaboration, re-planning, and decision-making. For example, intelligent and dynamically interactive software

modules that are responsible for pursuing the interests of instances of real world supply-chain objects, such as a particular requisition, a specific conveyance, or a single container, can achieve many of their objectives through employing services and engaging in negotiations that involve only a few nodes of the information management environment. This greatly reduces the propensity for the formation of communication bottlenecks and at the same time increases the amount of parallel activity in the computer-based environment.

The ability to combine in a computer-based information management environment many types of loosely coupled semi-autonomous and autonomous components (i.e., agents), representing a wide range of interests and incorporating different kinds of knowledge and capabilities, provides the environment with a great deal of versatility and potential for problem solving to occur simultaneously at several levels of granularity. This is similar to human problem solving teams in which individual team members work concurrently on different aspects of the problem and communicate in pairs and small groups as they gather information and explore sub-problems.

3.8 Emphasis on conflict identification

The capabilities of a computer-based logistic planning and execution environment should not be bound by the ultimate goal of automatic conflict resolution. Rather, the capabilities of the computing environment should support the identification of the conflict, presenting the human user with as much of the related context as possible. This notion gains in importance as the level of complexity of the logistic planning and management problem increases. The resolution of even mundane conflicts can provide subtle opportunities for advancing towards planning and/or execution objectives. These opportunities are more likely to be recognized by a human user than a computer-based agent. The identification of conflicts is by no means a trivial undertaking. It includes not only the ability to recognize that a conflict actually exists, but also the determination of the kind of conflict and the relationships and related context that describe the conflict and what considerations appear relevant to its resolution. The automatic tracing of these relationships may produce more progress toward a solution than the automatic resolution of the conflict itself.

3.9 Adaptability and agility

Traditionally, software tools categorized as intelligent were engineered for specific scenarios. Consequently, the successful application of these tools depended largely on the degree to which the characteristics of a particular problem component aligned with situations that the tool had been design for. This rigidity has tended to prove quite problematic when these tools were applied to even slight variations of the scenarios that they had been developed or trained for.

In contrast, what the experience of the author has shown is that intelligent tools not only need to support variation, but that these tools should be engineered with such adaptation as a core criterion. Much of this ability to effectively deal with variation is due to the ability of these tools to decompose complex problems into much more manageable components without losing the relationships that tie the components together. To accomplish this, the reasoning capabilities of the tools can be organized as discrete fragments of logic capable of addressing smaller components of the larger problem. If these components are described within an expressive, relationship-rich representation then the connections between the decomposed components are maintained automatically. The effects of addressing each individual component are automatically propagated across the entire expanse of the problem space due to the extensive set of relationships represented within the model that retains their connections and context. The result

is a problem solving tool that is agile in its ability to effectively adjust to the variable nature of the dynamically changing supply-chain.

3.10 The human-computer interface

The importance of a high degree of interaction between the human members of the supply-chain team and the various intelligent components of the computer-based information management environment is integral to most of the principles and requirements described above. This interaction is fundamentally facilitated by the information-centric representation core of the environment through which the interacting software components are able to maintain some level of understanding of the current context of the logistic planning and execution activities. However, there are other aspects of the user-interface that must be provided in support of the human-computer interactions. These include two-dimensional and three-dimensional graphical representation capabilities, explanation facilities, and a context-sensitive help system with semantic search support.

At a minimum the graphical capabilities must be powerful enough to include the accurate representation of the current geographical location and state of any transaction moving through the supply-chain, provide near real-time visual access to local conditions, support the animation of alternative movement plans, and allow past movements to be replayed. Technology permitting, the ultimate aim of an intelligent supply-chain environment is to provide a virtual reality user-interface that allows the human users to become fully immersed in the physical and emotional aspects of their logistic planning and execution activities.

Explanation facilities: The author's experience with decision-support systems over the past two decades has lent credence to the supposition that the need for an information management environment to be able to explain how it arrived at certain conclusions increases with the sophistication of the inferencing capabilities embedded in the software environment. At the very least, the intelligent components of the environment should be able to explain their results and methods of analysis. In this regard retrospective reasoning that is capable of providing answers to *what*, *how*, and *why* questions is the most common type of explanation facility available in multi-agent systems (Figure 3).

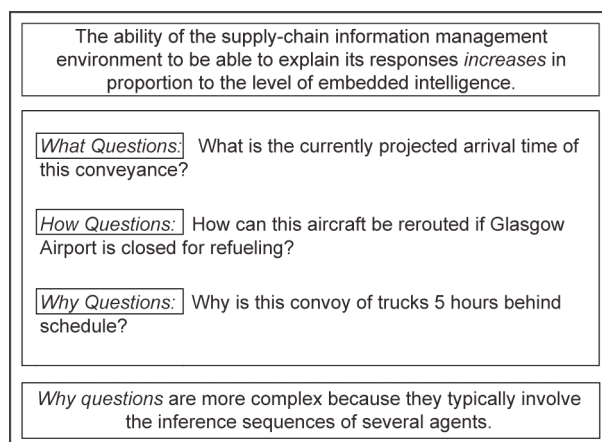


Figure 3: Explanation facilities

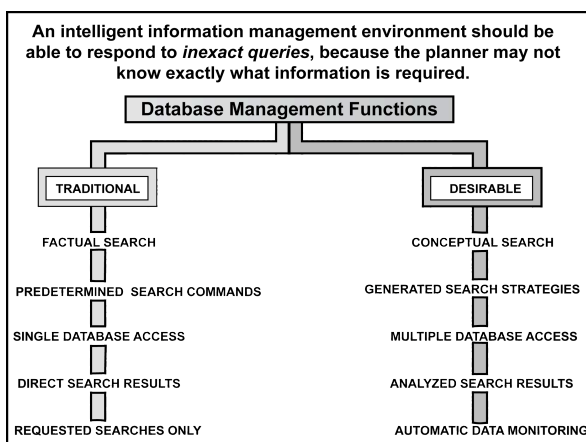


Figure 4: Semantic search facilities

A *what* question requires the explanation or definition of a fact. For example, the user may ask: *What is the currently projected arrival time of this aircraft and what is the certainty*

factor associated with this projection? In the past, expert system methodologies based on *format templates* would have allowed the appropriate answer to be collected simply through template values when a match is made with the facts (i.e., aircraft, departure time, wind conditions, etc.) contained in the question (Myers et al. 1993). Today, with the application of ontology-based reasoning capabilities more powerful and direct methods based on the ability of an ontology to represent concepts are available. A *how* question requires an analysis of the sequence of inferences or reasoning that produced the fact. Continuing with the above example, the user may ask: *How can this aircraft be rerouted if Glasgow Airport is closed for refueling?* The answer would require a sequence of inferences by the Fuel, Scheduling and Routing Agents.

Why questions are more complicated. They require reference to the sequence of goals that have driven the progression of inferences (Ellis 1989). For example: *Why is this convoy of trucks 5 hours behind schedule?* In large collaborative systems many agents may have contributed to the inference sequence and will need to participate in the formulation of the answer. This third level of explanation, which requires a summary of justification components, has received considerable attention over the past 30 years. For example: text summary systems such as Frump (Dejong 1982) and Scisor (Jacobs and Rau 1988); fast categorization techniques such as Construe (Hayes and Weinstein 1991); grammatical inference (Fu and Booth 1975) that allows inductive operators to be applied over the sequences of statements produced from successive justifications (Michalski 1983); explanation-based learning (Mitchell et al. 1991); and, case-based reasoning (Shank 1990 and 1991).

Semantic search facilities: While existing computer-based information management systems typically support only factual searches, an intelligent logistical planning and execution environment will provide semantic search capabilities that can deal with inexact queries (Figure 4). Due to the complexity of the problem space the human decision-makers will not always know exactly what information they require. Often they can define only in conceptual terms the kind of information that they are seeking. Also, they would like their query to be automatically broadened with a view to discovering additional information that may be relevant to their current problem solving focus.

The desirability of an information management environment to be able to deal with inexact search requests warrants further discussion. A flexible query capability, such as the human brain, can generate best guesses and a degree of confidence for how well the available information matches the query. For example, let us assume that the user is searching for a *pressure gauge* supply item. Before proceeding with the search the semantic query facility may ask the user to specify further search parameters such as measurement range, required accuracy, or type of fluid to be measured, and allow the user to enter a weighting factor to define the relative importance of each of those parameters that the user has been willing or able to specify. The result of the search would be a list of perhaps 10 *pressure gauge* type supply items ranked in order of probability of satisfying the user's query.

4. The Technical Approach

The desired capabilities of an intelligent logistical planning and execution environment outlined in the previous section call for a distributed system architecture that can be accessed from any physical location, is highly flexible, and totally transparent to the human user. In particular, the

user must be shielded from the many protocols and data and content exchange transformations that are required to access capabilities and maintain seamless interoperability among those capabilities. Any member of the supply-chain team, once authenticated during the single sign-on point of entry, should be able to access those capabilities (e.g., intelligent decision-assistance tools and data sources) that are included in the authentication certificate. The focus of the human user should not be on systems, as it still is mostly today, but on the capabilities or *services* that the computer-based environment can provide.

The notion of *services* is well established. Everywhere we see countless examples of tasks being performed by a combination of services, which are able to interoperate in a manner that results in the achievement of a desired objective. Typically, each of these services is not only *recomposable* but also sufficiently *decoupled* from the final objective to be useful for the performance of several somewhat similar tasks that may lead to quite different results. For example, a common knife can be used in the kitchen for preparing vegetables, or for peeling an orange, or for physical combat, or as a makeshift screwdriver. In each case the service provided by the knife is only one of the services that are required to complete the task. Clearly, the ability to design and implement a complex process through the application of many specialized services in a particular sequence has been responsible for most of mankind's achievements in the physical world.

4.1 Service-oriented architecture (SOA)

In the software domain these same concepts have gradually led to the adoption of Service-Oriented Architecture (SOA) principles. While SOA is by no means a new concept in the software industry it was not until Web services became available that the principles of this concept could be readily implemented (Erl 2008, Brown 2008). In the broadest sense SOA is a software framework for computational resources to provide services to customers, such as other services or users. The Organization for the Advancement of Structured Information (OASIS)¹ defines SOA as a “... *paradigm for organizing and utilizing distributed capabilities that may be under the control of different ownership domains*” and “...*provides a uniform means to offer, discover, interact with and use capabilities to produce desired effects with measurable preconditions and expectations*”. This definition underscores the fundamental intent that is embodied in the SOA paradigm, namely *flexibility*. To be as flexible as possible a SOA environment is highly modular, platform independent, compliant with standards, and incorporates mechanisms for identifying, categorizing, provisioning, delivering, and monitoring services.

The principal components of a conceptual SOA implementation scheme (Figure 5) include a Enterprise Service Bus (ESB), one or more portals to external clients with single sign-on facilities, and the enterprise services that facilitate the ability of the user community to perform its operational tasks.

Enterprise Service Bus (ESB): The concept of an Enterprise Service Bus (ESB) greatly facilitates a SOA implementation by providing specifications for the coherent management of services. The ESB provides the communication bridge that facilitates the exchange of messages among services, although the services do not necessarily know anything about each

¹ OASIS is an international organization that produces standards. It was formed in 1993 under the name of SGML Open and changed its name to OASIS in 1998 in response to the changing focus from SGML (Standard Generalized Markup Language) to XML (Extensible Markup Language) related standards.

other. According to Erl (2008), ESB specifications typically define the following kinds of message management capabilities:

- *Routing*: The ability to channel a service request to a particular service provider based on some routing criteria (e.g., static or deterministic, content-based, policy-based, rule-based).
- *Protocol Transformation*: The ability to seamlessly transform the sender's message protocol to the receiver's message protocol.
- *Message Transformation*: The ability to convert the structure and format of a message to match the requirements of the receiver.
- *Message Enhancement*: The ability to modify or add to a sender's message to match the content expectations of the receiver.
- *Service Mapping*: The ability to translate a logical business service request into the corresponding physical implementation by providing the location and binding information of the service provider.
- *Message Processing*: The ability to accept a service request and ensure delivery of either the message of a service provider or an error message back to the sender. Requires a queuing capability to prevent the loss of messages.
- *Process Choreography and Orchestration*: The ability to manage multiple services to coordinate a single business service request (i.e., choreograph), including the implementation (i.e., orchestrate). An ESB may utilize a Business Process Execution Language (BPEL) to facilitate the choreographing.
- *Transaction Management*: The ability to manage a service request that involves multiple service providers, so that each service provider can process its portion of the request without regard to the other parts of the request.
- *Access Control and Security*: The ability to provide some level of access control to protect enterprise services from unauthorized messages.

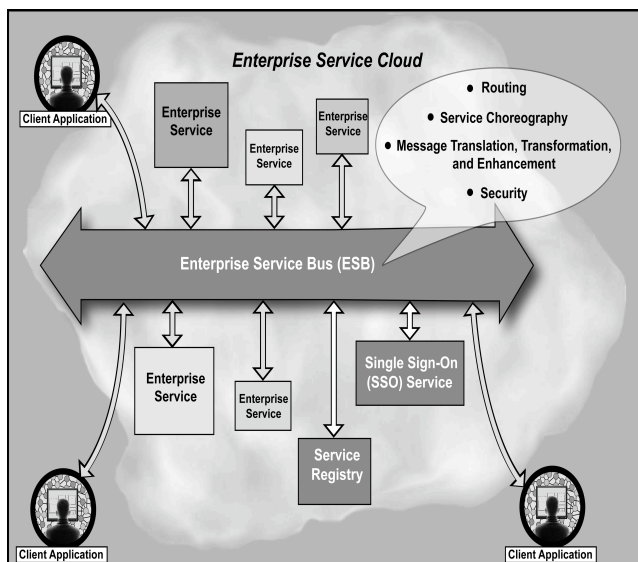


Figure 5: Principal SOA components

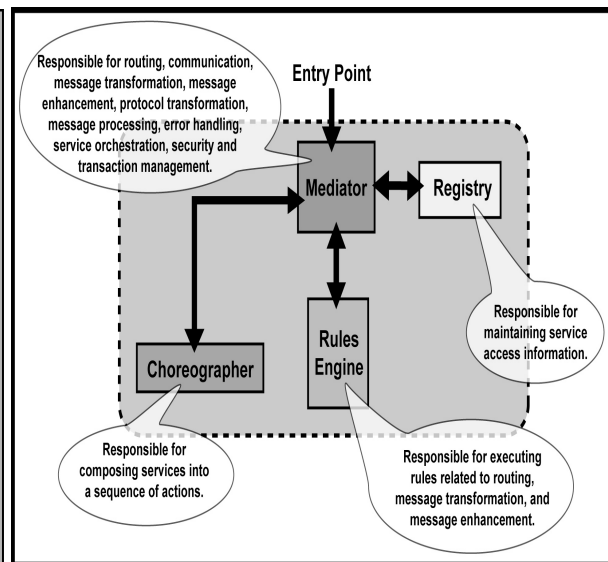


Figure 6: Principal ESB components

There are quite a number of commercial off-the-shelf ESB implementations that satisfy these specifications to varying degrees. A full ESB implementation would include four distinct components (Figure 6): Mediator; Service Registry; Choreographer; and, Rules Engine. The Mediator serves as the entry point for all messages and has by far the largest number of message management responsibilities. It is responsible for routing, communication, message transformation, message enhancement, protocol transformation, message processing, error handling, service orchestration, transaction management, and access control (security).

The Service Registry provides the service mapping information (i.e., the location and binding of each service) to the Mediator. The Choreographer is responsible for the coordination of complex business processes that require the participation of multiple service providers. In some ESB implementations the Choreographer may also serve as an entry point to the ESB. In that case it assumes the additional responsibilities of message processing, transaction management, and access control (security). The Rules Engine provides the logic that is required for the routing, transformation and enhancement of messages. Clearly, the presence of such an engine in combination with an inferencing capability provides a great deal of scope for adding higher levels of intelligence to an ESB implementation.

4.2 Information-centric representation

The methods and procedures that we human beings utilize to make decisions and solve problems rely heavily on our ability to identify, understand and manipulate entities, relationships, and related concepts. Such elements can be readily expressed in software as objects. In this respect, objects are complex symbols that convey meaning by virtue of the explicit and implicit contextual information that they encapsulate within their domain. For example, logistic planners develop shipment plans by reasoning about inventories, conveyances, routes, distribution centers, delivery windows, priority, weather, security, and so on. Each of these objects encapsulates knowledge about its own nature, its relationships with other objects, its behavior within a given environment, and the various constraints and requirements needed to effectively meet its individual performance objectives. This knowledge is contained in the various representational forms of the object as factual characteristics, algorithms, rules, and involvement in past scenarios (whether successful or problematic).

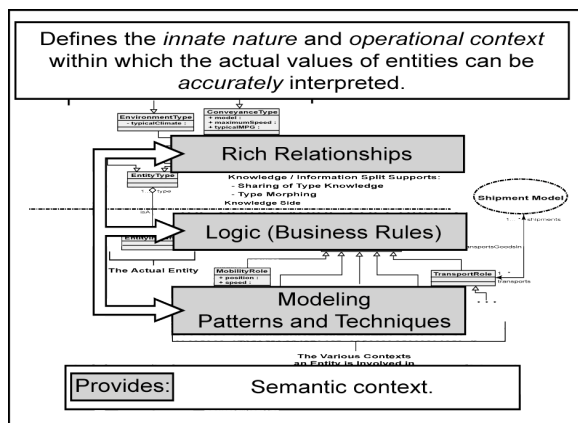


Figure 7: Ontology representation characteristics

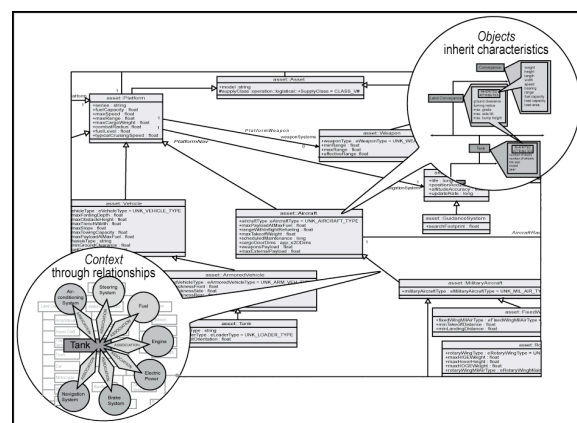


Figure 8: Ontology objects and concepts are machine processable

It is therefore apparent that a critical requirement for effective human-computer interaction in an intelligent supply-chain information management environment is the effective representation of

the *context* within which the logistic planning and management activities are taking place. This can be accomplished utilizing an *ontology* (Figure 7). The term ontology is loosely used to describe an information structure that is rich in relationships and provides a virtual representation of some real world environment. As shown in Figure 8, the elements of an ontology include objects and their characteristics, different kinds of relationships among objects, often including the concept of inheritance (Assal et al. 2009). To effectively align ontologies with the dynamics inherent within the real world, it is also important that a set of additional qualities be engineered into such models such as dynamic classification, multiple classification, incremental realization, and the ability to represent something that may not fit into any definition presently available. Since these elements of an ontology in combination with object-oriented computer languages (e.g., Java, C++) and advanced modeling paradigms (e.g., Web Ontology Language (OWL)) can be automatically interpreted by software, a computer-based information management environment can be endowed with at least a simplistic level of understanding of the real world *context* within which the required planning and execution decisions are being made. This level of understanding is sufficient to provide the necessary context for software agents to automatically interpret data, develop and evaluate plans, detect and explain the causes of conflicts, and generate warnings and alerts.

While an ontology is expressed in object-oriented terms, it is more than an object model. It is designed to describe the entities, concepts, and related semantics of some subject matter domain. Software that incorporates an internal information model, such as an ontology, is often referred to as *information-centric* software. The information model is a virtual representation of the real world domain under consideration and is designed to provide adequate *context* for software agents (typically rule-based) to reason about the current state of the virtual environment.

4.3 Software agents as intelligent tools

On the assumption of an information-centric software architecture that incorporates an ontology-based high level representation of the logistic planning and execution context, the intelligence of the information management environment is largely contributed by the inferencing tools that are available to the human user. Most of these tools will be in the form of invocable services or self-initiating agents. There is a behavioral distinction between services and agents. Services are invoked to perform a discrete activity, returning to their original inactive state after the activity has been completed. Agents on the other hand may be active on a continuous basis, taking the initiative opportunistically whenever they determine that the situation warrants an action. Often these agent actions will invoke services.

There are many types of software agents, ranging from those that emulate symbolic reasoning by processing rules, to highly mathematical pattern matching neural networks, genetic algorithms, and particle swarm optimization techniques. While all of these have capabilities that are applicable to an intelligent supply-chain environment, the symbolic reasoning agents will normally play the most important role and bring the most immediate benefits when a virtual context model (i.e., ontology) has been constructed. Therefore, only symbolic reasoning agents that can interact directly with the ontology-based context model will be discussed in this paper. For these rule-based agents the reasoning process relies heavily on the rich representation of entities and concepts provided by the ontology.

In general terms software agents with symbolic reasoning capabilities may be defined as tools that are situated, autonomous, and flexible (Wooldridge et al. 1999, Wooldridge 1997). They are

situated since they receive a continuous flow of operational information generated by the activities within and peripheral to the problem domain environment, and perform acts that may change that environment (e.g., creating alerts, making suggestions, and formulating recommendations). Agent tools are autonomous because they act without the direct intervention of human users, even though they allow the latter to interact with them at any time. In respect to flexibility, agent tools possess the three qualities that define flexibility within the context of the above definition. They are responsive, since they perceive their environment through an internal information model (i.e., ontology) that describes some of the entities and concepts that exist in the real world environment. They are proactive because they can take the initiative in making suggestions or recommendations. They are social, since they can collaborate with other agents or human users, when appropriate, to complete their own problem solving and to help others with their activities.

One important aspect of autonomy in agent applications is the ability of agents to perform tasks whenever such actions may be appropriate. This requires agents to be opportunistic, or continuously looking for an opportunity to execute. In this context opportunity is typically defined by the existence of sufficient information. For example, as the Weather Agent communicates an alert that a particular airport has been closed for the next six hours due to fog, several agents may become involved automatically to undertake analyses (e.g., rerouting alternatives, priority changes, contingency modifications) appropriate to their capability domains.

Service Agents: Agents that are designed to be knowledgeable in a specific domain, and perform planning or assessment tasks in partnership with other agents (i.e., human agents or software agents) are often referred to as Service Agents (Durfee 1988, Durfee and Montgomery 1990, Pohl et al. 1997). The manner in which they participate in the decision-making activities depends on the nature of the situation. Service Agents can be designed to react to changes in the problem state spontaneously through their ability to monitor information changes and respond opportunistically.

In an intelligent supply-chain information management environment Service Agents have knowledge and analysis capabilities in narrow logistic-related domains such as inventory assessment, fuel consumption, scheduling, weather data interpretation, cargo staging, terrain analysis, and maintenance. Typical analysis and inferencing characteristics of Service Agents include:

- Ability to generate alerts based on current state analysis.
- Ability to justify alerts, and analysis results with explanation facilities.
- Ability to broadcast requests for services to other agents.
- Ability to automatically generate queries and access data repositories.
- Ability to temporarily clone themselves to process multiple requests in parallel.
- Ability to undertake proactive explorations opportunistically.

Typical examples of Service Agents for logistical planning and management are described in Appendix A.

Planning Agents: Planning is a reasoning activity that deals with the availability of resources and the actions that need to be taken to complete a given task. Consequently, Planning Agents are designed to reason about the problem state and produce a plan based on the current state of the supply-chain in conjunction with the applicable constraints and objectives. This planning process involves matching the latter with the available resources to

produce a course of action that will satisfy the desired objectives. The complexity of the process can be reduced by distributing the basic planning tasks among a set of agents, as follows: identify the constraints and objectives; identify the available resources; note the unavailability of resources; identify the available set of actions or characteristics; and, generate a plan for satisfying the objectives.

Plan or solution generation is the actual planning activity in the above list of tasks. Many planning systems use specialized search algorithms to generate plans according to given criteria (Blum and Furst 1997). Re-planning, which is also commonly referred to as continual planning and includes dynamic planning, involves the re-evaluation of parts of an existing plan or solution because of a change in the information that has been used in the creation of that plan. Some planning systems take advantage of the feedback obtained from the monitoring and execution of plans to add to their knowledge by employing learning techniques, such as explanation-based learning, partial evaluation, experimentation, automatic abstraction, mixed-initiative planning, and case-based reasoning. There are several approaches to learning in agents, including reinforcement learning, classifier systems, and isolated concurrent learning. Learning techniques also enhance the communication ability of agents (Sen et al. 1994, Veloso et al. 1995).

In a supply-chain environment logistic Planning Agents deal with broader issues that relate to the ability of the shipping plan to meet customer requirements within planning and execution constraints such as the availability of inventory, conveyances, routes, and fuel, as well as delivery windows, cost, and acceptable risk. Typical analysis and inferencing characteristics of Planning Agents include:

- Ability to task Service Agents and request information from Mentor Agents.
- Ability to orchestrate evaluations involving several Service Agents.
- Ability to generate broad current state assessments on request or by alert.
- Ability to act on directions from human users and Coordination Agents.

Typical examples of Planning Agents for logistical supply-chain functions such as route planning, cost estimating, risk assessment, efficiency measurement, and opportunity recognition are described in Appendix B.

Mentor Agents: The purpose of a Mentor Agent is to temporarily provide a passive data element with active capabilities such as communication and limited self-determination (Pohl 1996). Mentor Agents are created either by human users or by Coordination Agents on a temporal basis to track a particular supply-chain object such as a requisition, container, pallet, or conveyance that is of special interest. In this way the instance of an object represented in the context model (i.e., ontology) is empowered to play an active role during its life cycle within the supply-chain (Figure 9).

The concept of Mentor Agents brings several potential benefits. First, it increases the granularity of the active participants in the problem solving process. As agents with collaboration capabilities, agentified data elements can pursue their own objectives and perform a significant amount of local problem solving without repeatedly impacting the communication and coordination facilities utilized by the higher level components of the distributed system. Typically, a Mentor Agent is equipped with communication capabilities, process management capabilities, information about its own nature, and objectives. Second, the ability of Mentor Agents to task Service Agents greatly increases the potential for

concurrent activities. Multiple Mentor Agents can request the same or different services simultaneously.

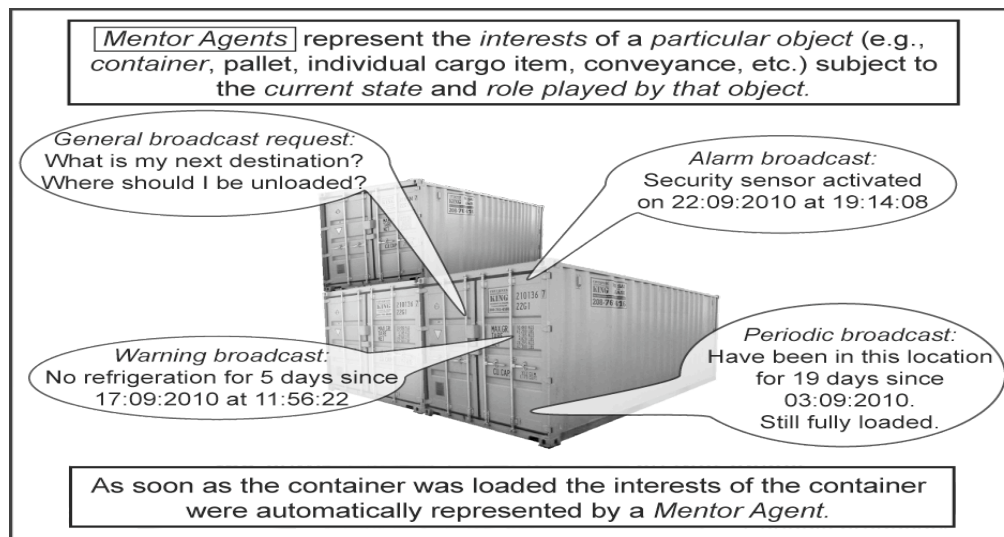


Figure 9: Mentor Agent representing a particular *container* in a shipment

Third, groups of Mentor 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 Mentor Agents are able to maintain their relationships to other aspects of the current state of the supply-chain. In this respect they are the product of *decentralization* rather than *decomposition*. In other words, the concept of Mentor 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.

In summary, the capabilities of a Mentor Agent that is created in support of the logistical tasks in an intelligent supply-chain environment would normally include one or more of the following:

- Some understanding of its needs as derived from the context model (i.e., ontology).
- Ability to orient itself geographically and geometrically (i.e., location).
- Ability to communicate and request services from Service Agents.
- Ability to communicate and negotiate with other Mentor Agents.
- Ability to pursue interests proactively leading to alternative recommendations.

Coordination Agents: This group of agents is responsible for facilitating collaboration among human users and software agents. Consequently Coordination Agents require the most intelligence because they need to be able to assess the impact of decisions in individual domains on the particular course of action under consideration (e.g., shipment plan), as well as the overall problem space (e.g., transportation network model).

Particularly in a logistic planning and management environment the most important and demanding role of Coordination Agents is to facilitate collaboration by activating agents and alerting human users of the need for interaction. This requires a relatively high level of understanding of the current state of the supply-chain, which can be only partially fulfilled by currently available artificial intelligence methodologies. Under these circumstances the ability of the human user to assist a Coordination Agent can bridge some of the machine intelligence challenges such as the representation and validation of knowledge that continue to plague the field of machine learning (Forsyth 1989, Thornton 1992, Johnson-Laird 1993). Accordingly Coordination Agents have a greater need than any of the other agent groups to interact with the human agents in the supply-chain information management environment. Through this interaction the human user will be able in several different ways to assist a Coordination Agent by contributing information and knowledge in a collaborative manner. Such human-based assistance may include the setting of priorities, the selection of a particular conflict resolution strategy, the directed invocation of specific agents, or the rejection of certain agent generated recommendations.

Another important function of Coordination Agents is the recognition of conflicts. The emphasis here is on the detection and identification of the causes of a conflict by the agent rather than its resolution. The resolution of a conflict usually involves higher level decisions that have the potential for impacting other areas of the supply-chain. Therefore, apart from very mundane conflicts that could be resolved automatically, the human user should at least be provided with an opportunity to resolve conflicts with wider consequences.

Typical examples of Coordination Agents for logistical supply-chain functions such as collaboration, conflict detection and analysis, threat assessment, and the identification of multi-modal (i.e., air, ship, rail, and truck convoy) transportation alternatives are described in Appendix C.

Governance Agents: While Governance Agents play a particularly important role in military logistic operations, they also have relevance to commercial supply-chains. In both the military and commercial domains these agents are concerned with the measurement of performance, the prevention of security breaches (i.e., theft in the commercial domain), the monitoring of priorities, and the identification of supply-chain trends. Specifically in the military domain, apart from these general functions, Governance Agents are also responsible for ensuring that individual shipment plans are in compliance with Commander's Intent, applicable Rules of Engagement (ROE), and force protection policies.

The role of Governance Agents to identify trends warrants further discussion. The detection of supply-chain trends is almost exclusively considered to be a human role in existing logistical planning and management networks. As a result, due to the large number of transactions that are involved in sizable supply-chains and the dynamically changing nature of the execution phase of operations, many opportunities for proactive planning are overlooked. Particularly under surge conditions in military operations, or when unforeseen events seriously disrupt shipment plans in either the military or commercial domain, the human decision-maker is forced into a reactive role. Unfortunately, it is not uncommon for these disruptions to be either considered one-time incidents that are unlikely to be repeated in the future or for the collection of lessons-learned to be neglected due to human exhaustion. In many cases, the existence of patterns that would, if recognized, lead to operational changes

with attendant efficiency improvements and cost savings are not readily discernable without continuous analysis over time.

Governance Agents with access to pattern matching tools such as neural networks can provide powerful trend detection capabilities. Since such tools are able to operate unobtrusively in background on a continuous basis they are able to address the following kinds of questions that are of interest at the executive level of supply-chain management:

- What quantity of any particular commodity or class of supplies (i.e., in the military domain) has been delivered to a specified geographic region or location over a given time period?
- What were the principal choke points where shipments have been delayed during a given time period?
- What has been the average time that certain kinds of shipments have taken over a given time period?
- What have been the relative densities of air, ship, rail, and truck movements over a given time period?
- What have been the principal causes of inter-modal delays or substitutions over a given time period?

Typical examples of Governance Agents for both military and commercial supply-chain functions are described in Appendix D.

4.4 The system environment

Conceptually, as shown in Figure 10, the logistical context provided by the multi-layered ontology allows the various groups of agents to monitor and act on the data that flows on a continuous basis through the supply-chain. The primary functions of the Planning Agents are focused on the generation of alternative route plans when needed and the determination of closure when a shipment has been delivered. However, the evaluation of these plans may also involve cost estimating, risk assessment, and the identification of opportunities for improving efficiency and reducing costs. The Coordination Agents are responsible for facilitating collaboration, exploring the availability and suitability of conveyances and arranging multi-modal movement plans. For example, if the Opportunity Agent identifies a partially loaded conveyance then the Collaboration Agent will immediately explore the possibility of backfilling this conveyance with another shipment to the same destination. This exploration may involve one or more Service Agents such as the Scheduling Agent and the Staging Agent to determine whether the existing schedule and staging plan of a candidate shipment can be modified to take advantage of the opportunity.

What is significant is that all of these actions can be undertaken automatically and concurrently for hundreds of shipment plans on a continuous near real-time basis. When events that have the potential for disrupting the supply-chain occur the human users have the necessary tools and actionable information available to take immediate and effective action. At the same time the Governance Agents are systematically analyzing past shipments with a view to identifying patterns and trends within the supply-chain. The purpose of this after-action analysis is to provide a basis for contingency planning and proactive actions that are aimed at reducing risk with attendant increases in efficiency and cost reductions in future transactions.

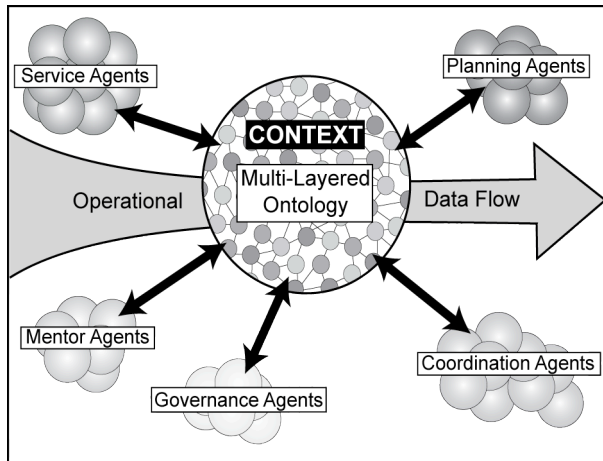


Figure 10: Context-based intelligent tools

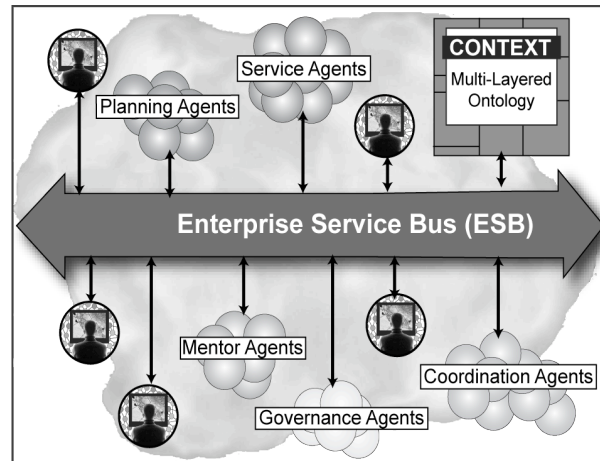


Figure 11: SOA-based system architecture

The system implementation framework is based on SOA principles (Figure 11), with interaction among the various loosely coupled applications and services managed transparently to the human users by an ESB. While many of the agents operate concurrently in an opportunistic mode, the workflow of logistical operations is essentially sequential in character. In a SOA-based system environment the orchestration of such sequences is normally performed by a Business Process Management (BPM) facility.

Business Process Management (BPM): BPM is a method for actively defining, executing, monitoring, analyzing, and subsequently refining manual or automated business processes. In other words, a business process is essentially a sequence of related, structured activities (i.e., a workflow) that is intended to achieve an objective or larger task. Such workflows can include interactions between human users, software applications or services, or a combination of both. In a SOA-based information management environment this orchestration is most commonly performed by the Choreographer component of the ESB (Figure 6). Based on SOA principles, a sound BPM design will decompose a complex business process into smaller, more manageable elements that comply with common standards and reuse existing solutions.

The principal components of the BPM capability within the supply-chain information management environment include a Business Process Execution Language (BPEL) engine, a graphical modeling service, business user and system administration interfaces, internal and external system interactions, and persistence. The BPEL is normally XML-based² and event driven. The BPEL engine is responsible for systematically issuing the sequence of service and/or user requests that are specified within the specific BPEL script, elegantly handling any related events or issues as they may occur.

While BPM and SOA concepts are closely connected, they are certainly not synonymous. Rather, they are complementary. Described more precisely, a SOA-based system environment provides the enabling infrastructure for BPM by separating the functional execution of the business process from its technical implementation. Conversely, BPM offers even the most well architected inventory of SOA functionality (i.e., services) specific objectives. The business

² The Extensible Markup Language (XML) is a general purpose specification that allows the content of a document to be defined separately from the formatting of the document.

process models identified as part of the BPM approach prove to effectively align the software capabilities produced to the actual needs of the users. Too often enterprises suffer from a distinct mismatch between available software functionality and actual user needs.

In addition to those components discussed above, an effective logistics decision-support environment includes a number of other principal components including:

- A web-based application portal that provides the human user with an integrated, highly-interactive canvas (i.e., view) across what may otherwise be a disparate collection of services, information sources (e.g., GIS, databases, etc.), intelligent agents, and external systems. Further, benefiting from the strong presence of BPM principles and functionality complementing the overarching SOA-based enterprise, this rich user interface is purposefully organized around the very business processes that are relevant to the specific type of user (e.g., logistics planner tasked with filling supply orders in an informed and efficient manner, tactical commander (in the military domain) wishing to verify the status of expected supplies, etc.). In other words, orienting the various flavors of the user-interface around relevant business processes provides specific users with a graphical, highly-interactive (essentially customized) user-interface that is designed and engineered in terms of the very workflows, terminology, and practices that comprise that user's tasks, objectives, and practices (i.e., business processes). The result is a convenient, highly efficient control panel that fosters an effective partnership between the human users and the software capabilities designed to assist them.
- An ontology service that builds, maintains, and exposes its evolving context to agents and other services that are context-dependent. Such informational services can support synchronization of interested clients with changes occurring within the context they manage via asynchronous service requests that can live for extended periods of time. The result is a means by which clients can subscribe to, and consequentially be notified of, particular events and conditions of interest as they may occur.
- An inference service that may comprise a number of agent communities. An agent community is a collection of related agents in a given domain such as the Planning Agents, Mentor Agents, Service Agents, Coordination Agents, and Governance Agents described in Section 4.3. Each agent utilizes applicable ontology services and other types of services to examine and analyze the current state of a particular transaction sequence or larger supply-chain context.

4.5 The user environment

From the human user's point of view the intelligent logistic planning and execution environment described in this paper is highly interactive and proactive. Not only are the users able to conduct searches for data where the search keys are known (i.e., directed searches) but they are also able to conduct semantic searches when the queries can be only vaguely formulated. In those cases agents with data mapping capabilities will search through one or more databases and return to the user approximately matching query results with computed certainty factors.

At the same time the user is automatically alerted to both opportunities for taking advantage of events that could lead to greater efficiency or lower shipment costs and events that either are already or could potentially disrupt the supply-chain. Since agents are continuously monitoring most aspects of the shipment traffic within the transportation network many of the opportunities for effective intervention that are likely to be overlooked in current data-centric management

systems will be brought to the attention of the human user through agent warnings and alerts. In this respect the intelligent logistic planning and execution environment is both reactive and proactive. For example, if any particular shipment is running behind schedule then this will be noted and recorded in a *warning* report by an agent. If a shipment is halted by an obstacle in its path such as traffic congestion, a flooded road or a fogged-in airport then this will be noted by an agent and the user will receive an *alert*. However, agents are also continuously analyzing past shipments to identify patterns and trends, so that these can be related to current or expected near term conditions within the transportation network. This type of analysis may involve multiple Governance, Coordination, Planning, and Service Agents, with the objective of identifying potential supply-chain events and disruptions proactively. For example, the repeated late delivery of shipments in a particular region may suggest the need for considering an alternative inter-modal movement plan.

Data access: Much of the management time in a supply-chain environment is spent on determining the location and status of shipments that have failed to arrive at their destinations within the time windows expected by the requesters. The logistical planning and execution environment must therefore provide in-transit visibility capabilities. These capabilities come as a by-product of the ontology-based context model that treats most of the graphical elements that are displayed in the user-interface on geographical maps as objects with characteristics and relationships. This allows the human user to lodge queries about a particular shipment or group of shipments and pursue such queries to reasonable depth, with the objective of receiving answers to the following kinds of questions: Where is this shipment right now? Where was the shipment last reported to have been seen or identified? What has been the event-by-event or node-to-node history of the shipment from the time it was first requested? What conveyances are available to expedite the movement of this shipment from where it is now to its intended destination?

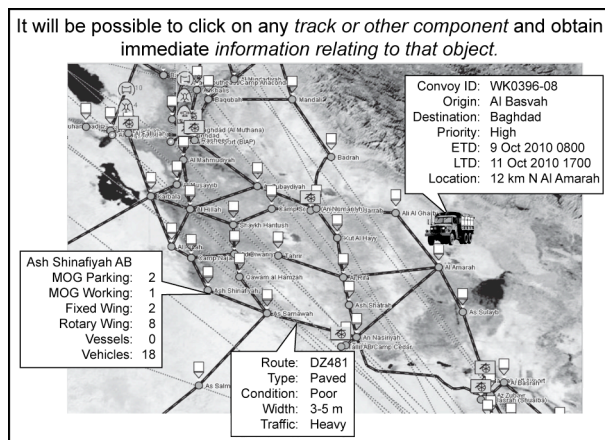


Figure 12: Displayed symbols are objects

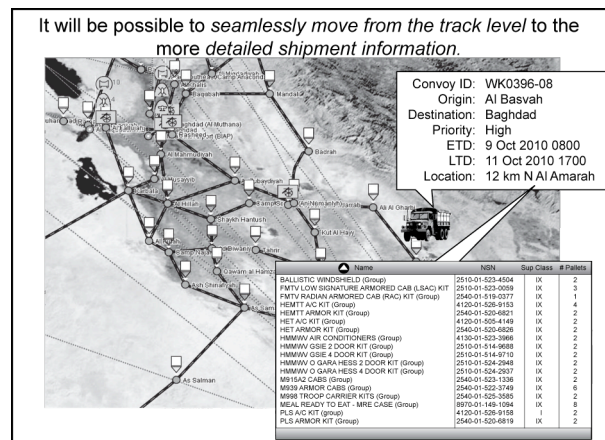


Figure 13: Information on request

As shown in Figure 12, to obtain information about any of the symbols displayed on the map the users simply clicks on the particular symbol (e.g., conveyance, supply center icon, city, or route) with their mouse. A second click allows a user to drill down to more detailed information. For example, in Figure 13 the user is able to seamlessly move from the summary information relating to the current location, destination, priority, and expected delivery window of a truck convoy, to the details of the individual cargo items.

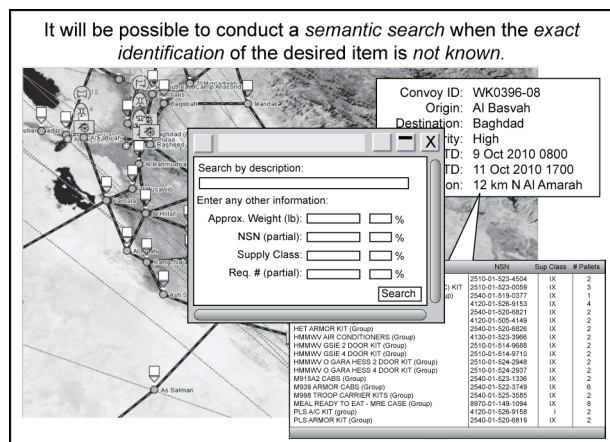


Figure 15: Search with partial information

Similarly, either by clicking on a displayed graphic symbol or by employing direct or semantic search capabilities the user is able to obtain a summary of the inventory of all of the supply centers in a particular geographical region (Figure 16) or drill down to the current inventory of a particular supply center (Figure 17). The same data is of course also available to agents based on automatically generated direct queries for use in the generation and evaluation of alternative plans, the assessment of risk, the determination of costs, and any other logistic management task that any particular agent is designed to perform.

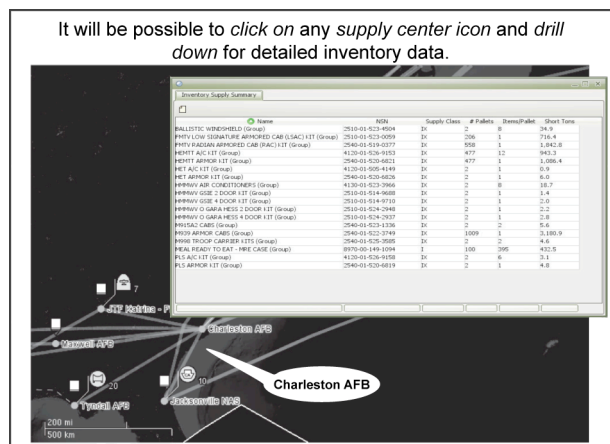


Figure 17: Supply center inventory details

96

- What does the track represent in terms of shipment ID, shipment type, and current transport mode (i.e., conveyance)?
- What is the last reported location of the track and what is the date and time of that location report?
- What is the next destination (i.e., node) of the track and what is/was the planned arrival date and time?

Similarly, the user is able to move seamlessly from the track level data to the more detailed shipment data, to answer questions such as:

- What is the priority of this shipment?
- What is the content of the shipment in terms of quantity and type of supplies?
- What was the origin of the shipment and the start date/time of the movement?
- What is the final destination of the shipment and who requested it? When was it requested? What was the requested delivery date/time? What was the delivery date/time according to the original movement plan? When is it most likely to be actually delivered?
- What is the node-to-node movement plan for this shipment? Where is it now in respect to this plan and what is the remaining unexecuted portion of the plan?

Impact of external factors: Both the formulation and execution of shipment plans is impacted by external factors such as weather conditions, customs requirements at border crossings or points of debarkation in foreign countries, location of criminal or enemy activities, availability of indigenous transportation, terrain, traffic conditions, and so on. In this respect an intelligent toolset is able to accept several on-line data feeds and combine the imported data with sufficient context to allow agents to automatically reason about the implications of the external factors. Candidate data feeds include:

- Weather forecasts on a regional and local level. For example, Figure 18 shows the translation of weather data by the Weather Agent into a weather report that provides actionable information to a human user and is machine processable for inferencing purposes by other software agents.
- Indigenous transportation systems (e.g., major roads, railways, ferries, commercial airline routes) in regions and local areas that may be available for shipments.
- Supplies, conveyances, fuel, and related transportation resources available at transportation hubs and distribution centers (Figure 19).
- Location of criminal and/or enemy activities.
- Infrastructure objects such as power plants, warehouses, railway stations, ferry stations, airports, ocean ports, fuel depots, and so on.

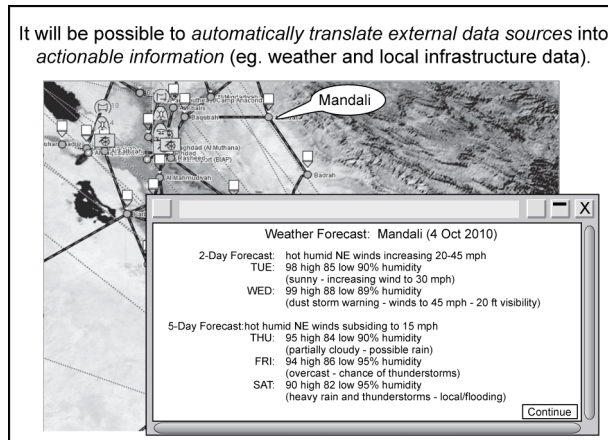


Figure 18: Weather report as actionable information for human and agent consumption

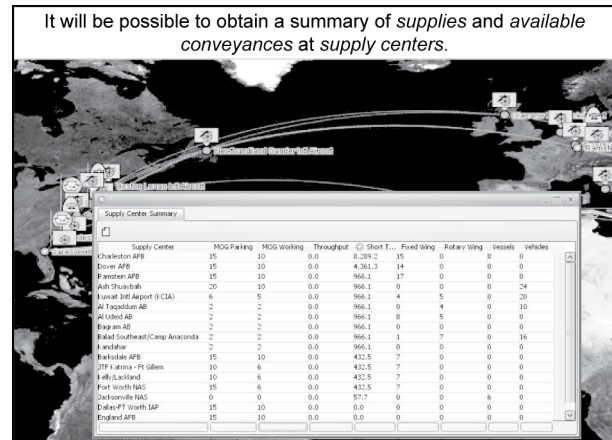


Figure 19: Distribution center inventory and available conveyances

Pattern recognition: As the scale of the adaptive toolset progressively encompasses a more significant portion of the supply-chain enterprise the intelligent agents will have access to an increasingly larger set of historical data. This will allow the implementation of agents with sophisticated analysis and case-based reasoning capabilities. Such agents, operating in a collaborative manner, will be able to analyze past shipments on a continuous basis and be able to respond to the following kinds of questions:

- What quantity of any particular kind of supplies has been delivered over a given time period, what shortages are likely to arise, and when?
- What were the principal choke points where shipments have been delayed during a given time period? Where are choke points likely to occur in the future based on current market forecasts?
- Where have shipments been intercepted by criminal or enemy action over a given time period and what are the risk factors that should be applied to future shipments?
- What has been the average time that certain kinds of shipments have taken over a given time period and how do these times relate to planned future movements?
- What have been the relative densities of air, surface and rail movements over a given time period and how do these densities relate to supply-chain performance?

4.6 Agent collaboration and decision-assistance

Historically, computer-based data-processing systems have been designed to be activated and controlled by human users. In this respect they may be characterized as *passive* decision-assistance environments that with few exceptions respond only when tasked by a human user. For example, the user enters the requirements for certain goods to be shipped between two geographical locations and a movement plan is either interactively formulated or automatically generated if more sophisticated tools are available. In other words, the user directs the system to assist in some predefined manner and the system generates the appropriate response or result to the best of its capabilities. If the users do not request the system to undertake any tasks then the system will be essentially idle.

A context-based (i.e., information-centric) software system with inferencing capabilities provided by agents is in contrast an *active* decision-assistance environment in which data cleansing, monitoring, analysis, planning and re-planning, pattern identification, and exploratory processing will occur on an on-going basis. In fact, under certain circumstances the system environment may be intensely active while the human users are largely inactive. The activities of the system environment are activated at least as much by the data that flows through the system on a continuous basis (Figure 10) as by the interactions of the human users with the system environment. This is largely made possible by the virtual model (i.e., multi-layered ontology) of the real world supply-chain context that allows the agents to autonomously and concurrently interpret and analyze the data flow in the appropriate context.

As an example of a typical sequence of logistical execution management events we will assume the following typical military scenario. A high priority requisition for add-on-armor (AOA) supplies comes to the Defense Logistics Agency (DLA) from Al Udeid in the Iraq theater and enters the Joint Deployment and Distribution Enterprise (JDDE) environment of the United States Transportation Command (USTRANSCOM).

As shown in Figure 20, the Priority Agent sends a *warning* to the Collaboration Agent suggesting that collaboration will be necessary due to the high priority of the request. The Collaboration Agent starts monitoring the requisition and immediately requests the Opportunity Agent to determine whether the requested AOA items are already in theater or in-transit to the theater. The Opportunity Agent invokes the Inventory Agent, which in turn seeks the assistance of the Distribution Center Agent and the Closure Agent to determine whether the requested AOA items are or will be available in the theater by the required date. Concurrently the Inventory Agent with the assistance of the Distribution Center Agent determines whether the required AOA items are in stock at a CONUS³ supply center.

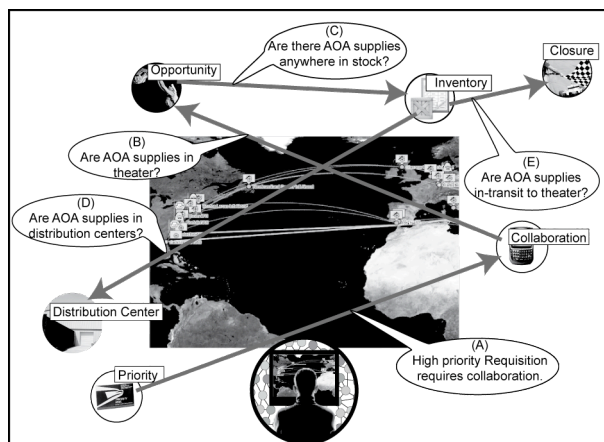


Figure 20: Are the requested AOA supplies available in inventory?

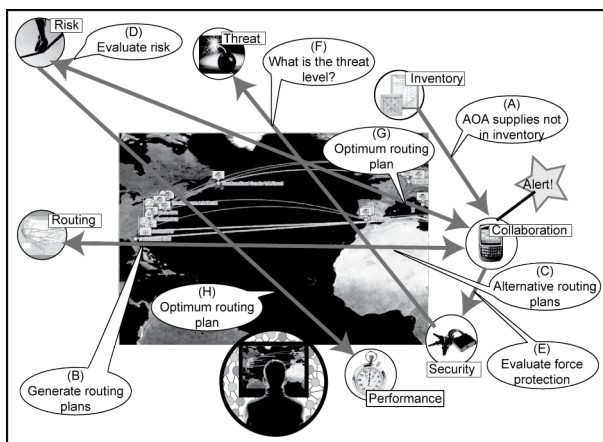


Figure 21: The supplies are not available and must be outsourced.

In Figure 21, the Collaboration Agent determines on the basis of the report received from the Inventory Agent that the requested supplies are not in CONUS inventory and decides to outsource to commercial supplier(s). Concurrently the Routing Agent is invoked by the

³ Continental United States (CONUS) includes the 48 states on the continent of North America that are south of Canada plus the District of Columbia, but excludes the states of Alaska and Hawaii, and all off-shore United States (US) territories and possessions.

Collaboration Agent to generate alternative multi-modal route plans from Charleston to Al Udeid and sends the plans to the Security Agent to address force protection issues and the Risk Agent to assess the risk of non-performance. The Security Agent requests the assistance of the Threat Agent in its analysis, while the Risk Agent shares the results of its analysis with both the Collaboration Agent and the Performance Agent.

In the meantime, the Collaboration Agent requests the creation of a Mentor Agent for this requisition (Figure 22). The Mentor Agent keeps track of all matters pertaining to this requisition such as: name of vendor; delivery window of AOA supplies to Charleston for shipping to Al Udeid.

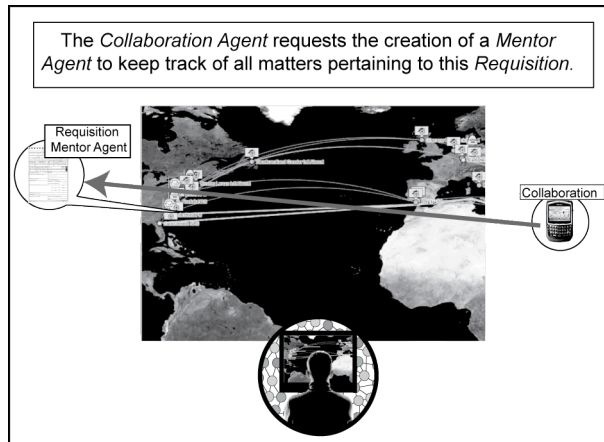


Figure 22: Mentor Agent is assigned to the high priority requisition

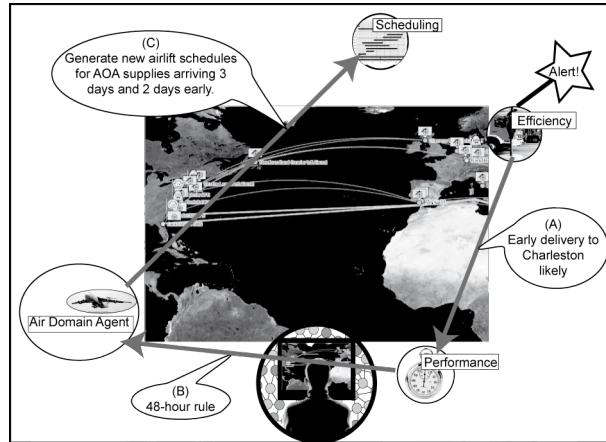


Figure 23: Potential Thanksgiving holiday build-up at Charleston POE⁴

In Figure 23, the Efficiency Agent notices that the delivery window for Charleston is 22-24 November, which is just before the Thanksgiving holiday. It therefore sends an *alert* to the Performance Agent indicating that early delivery to Charleston by commercial shippers to accommodate personal holiday plans is likely to cause a build-up of shipments at Charleston. The Performance Agent being aware of the 48-hour rule that does not allow cargo to be staged at Charleston for longer than 48 hours prior to shipping, sends a *warning* to the Air Domain Agent. The latter proactively requests alternative schedules from the Scheduling Agent based on most (i.e., 80%) of the AOA cargo arriving at Charleston 3 days and 2 days before Thanksgiving.

Continuing in Figure 24, the Air Domain Agent determines on the basis of the schedules generated by the Scheduling Agent that the airlift assets available at Charleston will be inadequate and sends an *alert* to the Collaboration Agent. In Figure 25, the Collaboration Agent requests shipping cost estimates based on early and late purchase orders from the Cost Agent and then sends an *alert* to the human user to the likely requirement of commercial airlift with the cost estimates in hand. In the meantime, the Risk Agent assesses the risks involved in early and late purchase decisions. The human user decides on the basis of the high priority of the shipment, and the reports received from the Risk Agent and the Cost Agent that an early decision to order commercial airlift is warranted and approves the necessary purchase orders.

⁴ Point of Embarkation (POE).

It should be noted that the decision to place an immediate order for commercial airlift, thereby taking advantage of advance notice cost savings, has been made in minutes instead of days (or not until the need for commercial airlift has been noticed at the last moment by human users).

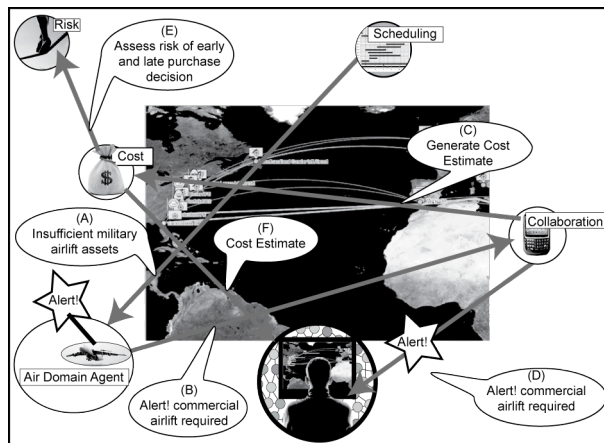


Figure 24: Early decision on commercial airlift required

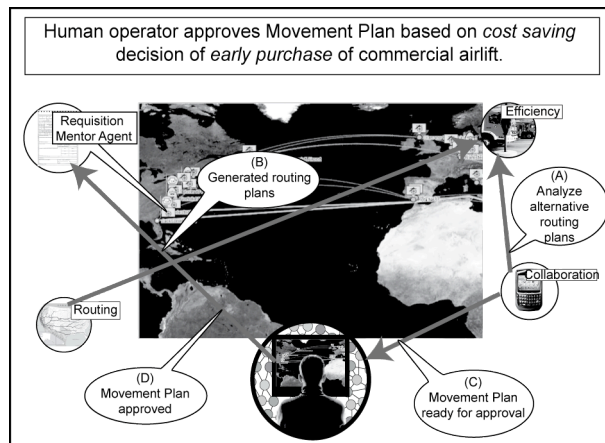


Figure 25: Decision to order commercial airlift made in minutes instead of days

Concurrently, in Figure 25, the Efficiency Agent is invoked by the Collaboration Agent to analyze the alternative plans generated by the Routing Agent, with the objective of determining the optimum movement plan. The human user approves the movement plan based on recommendations received from the Collaboration Agent. Again, recognition of the potential build-up of cargo at Charleston and the need for commercial airlift resources, as well as the decision to place an early purchase order and generate a new shipment plan all occurred in minutes.

By this time the Mentor Agent holds the following information about the requisition:

- Requisition ID, date received, ID of requesting party, and priority.
- Destination and requested delivery window.
- Name, NSN⁵, number of pallets, number of items per pallet, supply class, and weight of each requested AOA supply item.
- ID of commercial vendor for each outsourced AOA supply item.
- Force protection rating.
- Risk of non-performance rating.
- Estimated costs of supplies.

4.7 Execution scenario examples

During subsequent execution stages the Mentor Agent continues to look after the interests of the high priority requisition and the Collaboration Agent invokes any other agents to assist in the analysis and resolution of unforeseen events until the Closure Agent determines that the transaction has been completed.

The following two execution scenarios are not only typical of the military domain, but could equally well occur in a commercial supply-chain. The shipment plan approved by the human

⁵ National Stock Number (NSN).

user in Figure 25 includes Glasgow Airport in Scotland as a refueling venue. However, in its continuous monitoring and interpretation of global weather reports the Weather Agent discovers that Glasgow Airport is fog-bound. It immediately sends an *alert* to the Collaboration Agent indicating that Glasgow Airport is fog-bound (Figure 26). The Collaboration Agent requests the Routing Agent to generate an alternative movement plans with the assistance of the Air Domain Agent. Concurrently the Collaboration Agent requests the Efficiency Agent to analyze the alternative plans generated by the Routing Agent to determine an optimum alternative shipment plan. The Efficiency Agent receives input from the Cost Agent and the Security Agent during the analysis. Finally, the human user reviews the recommendations received from the Collaboration Agent and approves the new Movement Plan.

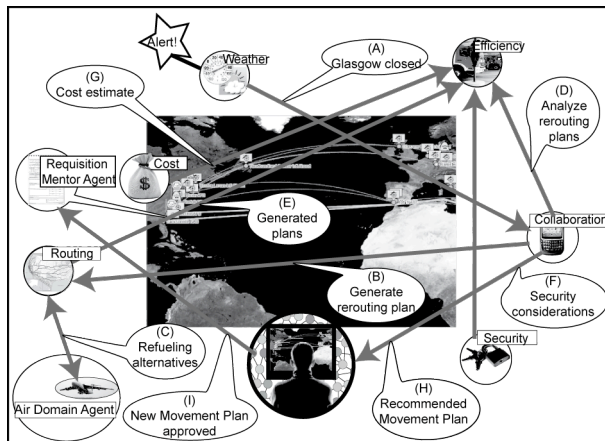


Figure 26: Glasgow Airport is fogged in and flights will need to be rerouted

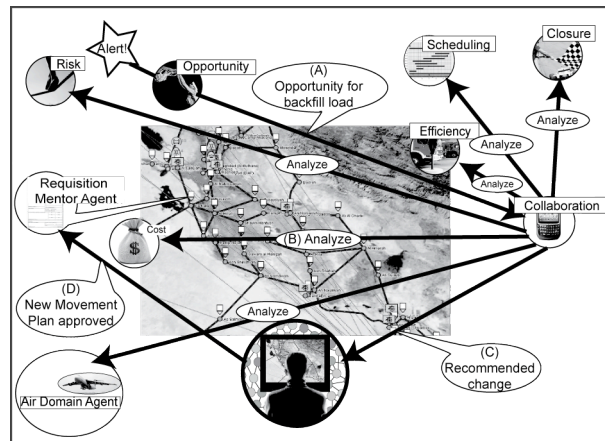


Figure 27: A backfill opportunity is not overlooked by the agents

The second example scenario deals with an opportunity to increase efficiency and reduce costs that would likely be overlooked by human users. Late arrival of another unrelated shipment to the same destination provides an opportunity for part of this shipment to backfill partial aircraft loads from Charleston to Al Udeid. In Figure 27, the Opportunity Agent sends an *alert* to the Collaboration Agent indicating an opportunity for saving transportation costs and time. It has discovered that due to late arrival at Charleston of some cargo from another requisition there may be a backfill opportunity. The Collaboration Agent immediately undertakes an analysis with the assistance of the Air Domain Agent, the Scheduling Agent, the Cost Agent, the Risk Agent, the Efficiency Agent, and the Closure Agent. The human user reviews the recommendations received from the Collaboration Agent and approves the modified shipment plan. Consequently, the Collaboration Agent informs the Convoy Domain Agent that part of the shipment for this requisition will be airlifted from the POE directly to Al Udeid and will therefore not require road transportation.

5. Conclusions

The inordinately high complexity of logistical planning and management tasks in a global supply-chain is due to the multitude of issues involved (e.g., routing, cost, risk, efficiency, security, priority, weather conditions, priority, inventory, conveyance type, terrain, and so on), the relationships among those issues, the frequency of changes during execution that threaten to disrupt the supply-chain, the time critical nature of shipments, and the diversity of the players

involved⁶. Management of this compound complexity requires the assistance of an intelligent software system environment (Figure 28).

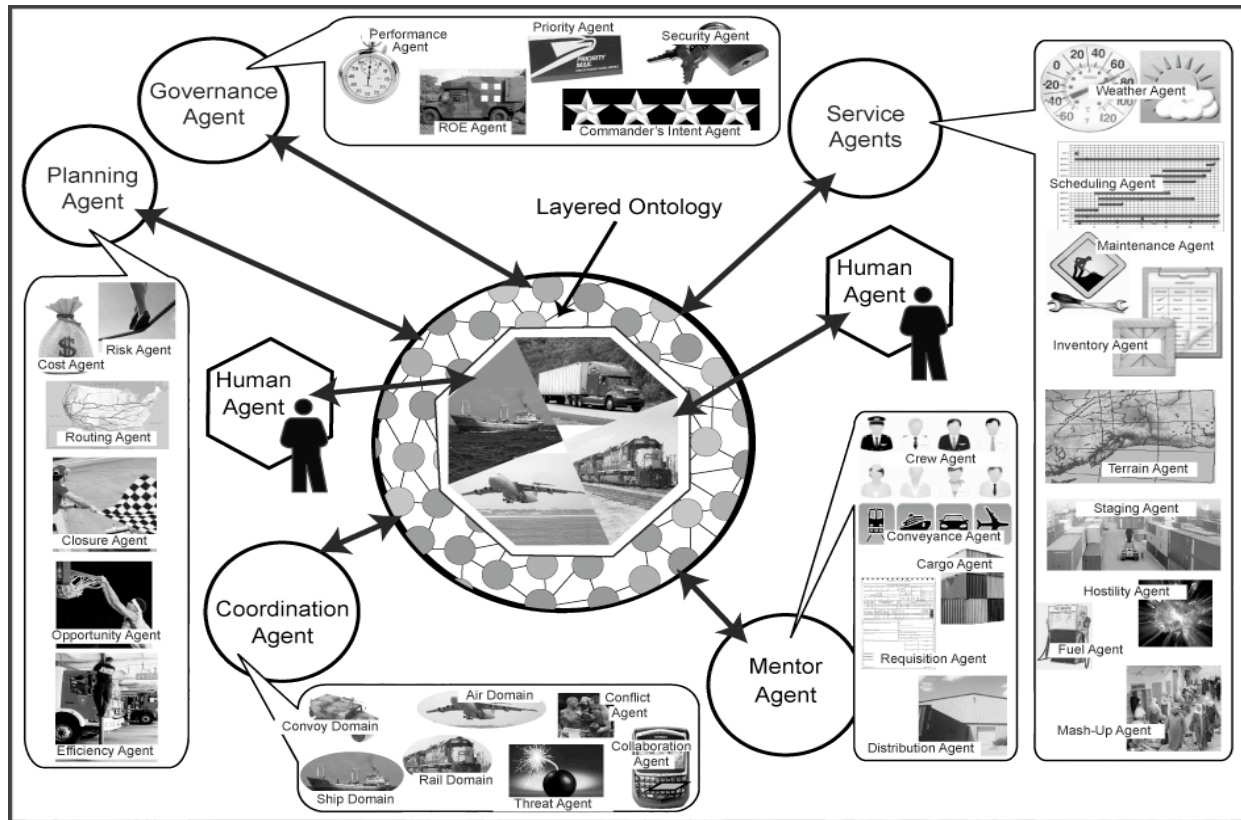


Figure 28: Enabling elements of an intelligent supply-chain information management system

As discussed in this paper there are two principal requirements for such an environment. The first requirement is a rich contextual representation of supply-chain information. This can be provided by a virtual model of the real world context within which the logistical management tasks such as the preparation of a multi-modal shipment plan, maintaining in-transit visibility, reacting to unforeseen events, preparing proactively for potential future events, and so on, can be expeditiously performed. The importance of this virtual model of real world context must not be underestimated. As a core requirement it provides the basis of most of the assistance capabilities of the intelligent information management environment described in this paper. Without access to the context provided by the multi-layered ontology the different groups of software agents defined in Section 4.3 and the Appendices could not function as intelligent tools in the manner described in Sections 4.6 and 4.7.

The second requirement is collaboration among the human users, as well as interaction between the human users and the intelligent software tools (e.g., agents) and, as discussed in Sections 4.6 and 4.7, between the intelligent tools themselves. Effective collaboration between any two parties assumes at least some commonality of purpose. Between human parties this commonality

⁶ The players or stakeholders in a supply-chain typically have very different objectives. For example, the planner is interested in high efficiency at minimum cost, the shipper is concerned about conveyance reliability and route conditions, while the customers expect to receive their orders on time and in an undamaged state.

is based not only on the understanding that each party has of its own objectives, but also on some level of understanding of the objectives and needs of the other party. In addition, there is a distinctly opportunistic aspect to collaboration. While the general requirement for collaboration and even the protocol that must be adhered to during the process of collaboration may be prescribed, the events that will initiate collaboration are largely unpredictable.

Similar principles of collaboration apply to the interactions between the human users and the software agents, and among the software agents themselves. The human users will expect the agents that they interact with to have some semblance of common understanding of the content of the interaction. This applies equally whether the user is requesting an explanation of an agent-generated result or queries the agent for specific information. Similarly, agents need some understanding of context to determine under what circumstances they should send an alert to human users or other agents. Clearly, the prerequisite for this semblance of understanding is the existence of a virtual model of real world context at the software level.

The current state of technology in software development provides the means for implementing a distributed, collaborative, intelligent, information management environment. Service-oriented architecture (SOA) concepts provide the framework and the guiding principles for developing distributed, service-based systems. The field of ontology representation is sufficiently mature to support the expressive modeling of domain knowledge as the enabling foundation for intelligent software tools or agents. Such agents can continuously monitor the supply-chain, participate in decision-making processes within specific domains, gather and present relevant information to the human user, and opportunistically communicate with human users and other agents.

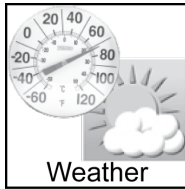
References

- AEDOT (1992); 'AEDOT Prototype 1.1: An Implementation of the ICADS Model'; Technical Report CADRU-07-92, Collaborative Agent Design Research Center, Cal Poly, San Luis Obispo, CA 93407.
- AMR Research (2007): <http://www.gartner.com/technology/supply-chain/amr-research.jsp>
- Assal H., K. Pohl and J. Pohl (2009); 'The Representation of Context in Computer Software'; Pre-Conference Proceedings, Focus Symposium on Knowledge Management Systems, InterSymp-2009, Baden-Baden, Germany, 4 August.
- Barber K., A. Goel, D. Han, J. Kim, D. Lam, T. Liu, M. MacMahon, C. Martin and R. McKay (2003); 'Infrastructure for Design, Deployment and *Experimentation* of Distributed Agent-based Systems: The Requirements'; The Technologies, and an Example, Autonomous Agents and Multi-Agent Systems. Volume 7, No. 1-2 (pp 49-69).
- Blum A. and M. Furst (1997); 'Fast Planning Through Planning Graph Analysis'; Artificial Intelligence, 90 (pp.281-300).
- Chartered Institute of Logistics and Transport in the UK (CILT) (1998); 'Members Directory'; CILT, Corby, www.ciltuk.org.uk.
- Brown P. (2008); 'Implementing SOA: Total Architecture in Practice'; Addison-Wesley.
- Dejong G. (1982); 'An Overview of the Frump System'; Lehnert and Ringle (eds.) Strategies for Natural Language Processing, Lawrence Erlbaum, Hillsdale, New Jersey (pp.149-176).
- Diaz C., W. Waiters, J. Pickard, J. Naylor, S. Gollery, P. McGraw, M. Huffman, J. Fanshier, M. Parrott, S. O'Driscoll-Packer, Boone Pendergrast and Evan Sylvester (2006); 'ICODES: Technical and Operational Description'; Technical Report CDM-20-06, CDM Technologies Inc., San Luis Obispo, California, November.
- Durfee E. (1988); 'Coordination of Distributed Problem Solvers'; Kluwer Academic, Boston, Massachusetts.

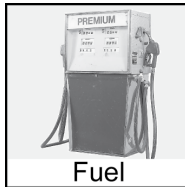
- Durfee E. and T. Montgomery (1990); 'A Hierarchical Protocol for Coordination of Multiagent Behavior'; Proc. 8th National Conference on Artificial Intelligence, Boston, Massachusetts (pp.86-93).
- Ellis C. (1989); 'Explanation in Intelligent Systems'; in Ellis (ed.) *Expert Knowledge and Explanation: The Knowledge-Language Interface*, Horwood, England.
- Erl T. (2008); 'SOA: Principles of Service Design'; Prentice Hall.
- Forsyth R. (1989); 'Machine Learning: Principles and Techniques'; Chapman and Hall, Computing Series, London, England.
- Fu K. and T. Booth (1975); 'Grammatical Inference: Introduction and Survey'; IEEE Transactions on Systems, Man, and Cybernetics. SMC-5: (pp.95-111, 409-423).
- Gero J., M. Maher and W. Zhang (1988); 'Chunking Structural Design Knowledge as Prototypes'; Working Paper, The Architectural Computing Unit, Department of Architectural and Design Science, University of Sydney, Sydney, Australia.
- Handfield R. (2008); 'Consumers of Supply Chain Risk Data'; in Handfield R. and K. McCormack (eds.) 'Supply Chain Risk Management: Minimizing Disruptions in Global Sourcing', Auerbach Publications, New York.
- Handfield R., J. Blackhurst, D. Elkins and C. Craighead (2008); 'A Framework for Reducing the Impact of Disruptions to the Supply Chain: Observations from Multiple Executives'; in Handfield R. and K. McCormack (eds.) 'Supply Chain Risk Management: Minimizing Disruptions in Global Sourcing', Auerbach Publications, New York (pp. 29-50).
- Hayes P. and S. Weinstein (1991); 'Construe-TIS: A System for Content-Based Indexing of a Database of News Stories'; Rappaport and Smith (eds.) *Innovative Applications of Artificial Intelligence 2*, AAAI Press, Menlo Park, California (pp.47-64).
- ICADS (1991); 'ICADS Working Model Version 2 and Future Directions'; Technical Report CADRU-05-91, Collaborative Agent Design Research Center, Cal Poly, San Luis Obispo, CA 93407.
- Jacobs P. and L. Rau (1988); 'A Friendly Merger of Conceptual Analysis and Linguistic Processing in a Text Processing System'; Proceedings of the Fourth IEEE AI Applications Conference, IEEE Computer Society Press, Los Alamitos, California (pp.351-356).
- Johnson-Laird P. (1993); 'Human and machine Thinking'; Erlbaum, Hillsdale, New Jersey.
- Manuj I, J. Dittmann and B. Gaudenzi (2007); 'Risk Management'; in Mentzer J., M Myers and T. Stank (eds.) 'Handbook of Global Supply Management', Sage Publications, London, UK (pp. 319-336).
- Michalski R. (1983); 'A Theory and Methodology of Inductive Learning'; Artificial Intelligence, Vol.20 (pp.111-161).
- Mitchell T., J. Allen, P. Chalasani, J. Cheng, O. Etzioni, M. Ringuette and J. Schlimmer (1991); 'Theo: A Framework for Self-Improving Systems'; VanLehn (ed.) *Architectures for Intelligence*, Twenty-Second Carnegie Mellon Symposium on Cognition, Lawrence Erlbaum, Hillsdale, New Jersey (pp.323-355).
- 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'; Technical Report CADRU-08-93, Collaborative Agent Design Research Center, Cal Poly, San Luis Obispo, CA 93407, January.
- Nibecker J., H. Larsen, X. Pan, C. Warren, R. Chambers, D. Taylor, B. Weber, J. Delos Reyes, C. Maas, and M. Porczak (2007); 'TRANSWAY: Technical and Operational Description'; Technical Report, CDM-21-07, CDM Technologies, Inc., San Luis Obispo, CA 93401, October.
- Pan J. and J. Tenenbaum (1991); 'Toward an Intelligent Agent Framework for Enterprise Integration'; Proc. Ninth National Conference on Artificial Intelligence, vol.1, San Diego, California, July 14-19 (pp.206-212).
- Pohl J. (2008); 'Cognitive Elements of Human Decision-Making'; in Jain L. and G. Wren (eds.); *Intelligent Decision Making: An AI-Based Approach*; Springer Verlag, New York (41-76).

- Pohl J., A. Chapman, K. Pohl, J. Primrose and A. Wozniak (1997); 'Decision-Support Systems: Notions, Prototypes, and In-Use Applications'; Technical Report, CADRU-11-97, Collaborative Agent Design Research Center, Cal Poly, San Luis Obispo, CA 93407, January.
- Pohl J. and L. Myers (1994); 'A Distributed Cooperative Model for Architectural Design'; in Carrara G. and Y. Kalay (eds.) Knowledge-Based Computer-Aided Architectural Design, Elsevier, Amsterdam, The Netherlands.
- Pohl K. (2008); 'A Translation Engine in Support of Context-Level Interoperability'; Special Issue on Ontology Driven Interoperability for Agile Applications Using Information Systems: Requirements and Applications for Agent Mediated Decision Support, Intelligent Decision Technologies, 2 (1), January.
- Pohl K. (2007); 'Enhancing the Face of Service-Oriented Capabilities'; Pre-Conference Proceedings, Focus Symposium on Representation of Context in Software, InterSymp-2007, Baden-Baden, Germany, 31 July.
- Pohl K. (1996); 'KOALA: An Object-Agent Design System'; in Pohl J. (ed.) Proc. Focus Symposium on Advances in Cooperative Environmental Design Systems, InterSymp-96, Baden-Baden, Germany, Aug.14-18 (pp.81-92), Collaborative Agent Design Research Center, Cal Poly, San Luis Obispo, CA 93407.
- Swaminathan J., S. Smith and N. Sadeh (1998); 'Modeling Supply Chain Dynamics: A Multiagent Approach'; Decision Sciences, 29(3), Summer (pp. 607-632).
- Schank R. and R. Osgood (1990); 'Content Theory of Memory Indexing'; Technical Report 2, The Institute for the Learning Sciences, Northwestern University.
- Schank R. (1991); 'Case-Based Teaching: Four Experiences in Educational Software Design';
- Sen S., M. Sekaran, and J. Hale (1994); 'Learning to Coordinate Without Sharing Information'; in National Conference on Artificial Intelligence (pp.426-431).
- Simon H. (1996); 'The Sciences of the Artificial'; 3rd ed., MIT Press, Cambridge, Massachusetts (pp. 111-3).
- Taylor D. and H. Assal (2009) 'Using BPM to Support Systems Interoperability'; The International C2 Journal, Vol. 3, No. 1.
- Thornton C. (1992); 'Techniques in Computational Learning'; Chapman and Hall, Computing Series, London, England.
- Veloso M., J. Carbonell, A. Perez, D. Borrajo, E. Fink and J. Blythe (1995); 'Integrating Planning and Learning: The PRODIGY Architecture'; Journal of Theoretical and Experimental Artificial Intelligence, 7(1).
- Waters D. (ed.) (2007); 'Global Logistics: New Directions in Supply Chain Management'; 5th edition, Kogan Page, Philadelphia.
- Wooldridge M., N. Jennings and D. Kinny (1999); 'A Methodology for Agent-Oriented Analysis and Design'; Proceedings Third International Conference on Autonomous Agents (Agents-99), Seattle, Washington.
- Wooldridge M. (1997); 'Agent-Based Software Engineering'; IEEE Transactions on Software Engineering, 144(1), (pp.26-37), February.

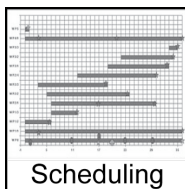
Appendix A: Typical Service Agents



1. The ***Weather Agent*** has the ability to interpret and translate raw weather data into a weather report that has meaning to both the human user and the computer (i.e., is machine processable)



2. The ***Fuel Agent*** has the ability to monitor the fuel consumption of conveyances during movements (through sensor data), project fuel requirements, locate refueling nodes, and assess the fuel capacity at nodes.



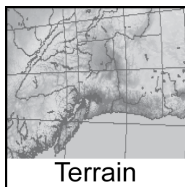
3. The ***Scheduling Agent*** is capable of integrating inter-modal movements, taking into consideration the delivery dates of cargo at the POE, the availability of surface and air transportation, and delivery windows.



4. The ***Staging Agent*** is capable of planning the staging of cargo in marshalling yards taking into account projected cargo arrival dates/times, order of loading based on conveyance type and destination, access routes, and space constraints.



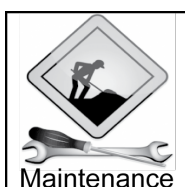
5. The ***Inventory Agent*** is responsible for monitoring the inventory of distribution centers and therefore has the ability to access data sources and formulate queries on an on-going basis, as well as in response to requests for inventory information from other agents and human users.



6. The ***Terrain Agent*** has the ability to assess the state of surface routes in terms of traffic congestion, impediments (e.g., flooded areas, land slides, snow, ice), road conditions and grades, and their potential impact on traveling time.



7. The ***Hostility Agent*** is responsible for monitoring potentially hostile activities that could impact shipments moving on surface routes, including theft, narcotics, piracy, terrorism, and enemy actions (in the military domain).



8. The ***Maintenance Agent*** is responsible for monitoring the maintenance requirements of conveyances and therefore has the ability to both access appropriate data sources and to monitor the operational state of

conveyances and high value loading facilities through the interpretation of sensor data.



9. The ***Mash-Up Agent*** is capable of generating a web application that combines data and/or existing Internet functionality (e.g., Google Earth) from multiple sources into an action report, such as an on-the-spot view of a local event (e.g., disaster area survey, cargo loading at an ocean port).

Appendix B: Typical Planning Agents



1. The ***Routing Agent*** has the ability to plan and re-plan multi-modal routing alternatives under time critical conditions, taking into consideration route conditions, efficiency, cost, and risk.



2. The ***Cost Agent*** has the ability to rapidly estimate the cost of alternative movement plans during both strategic planning and execution.



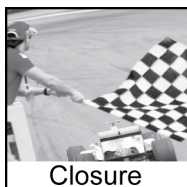
3. The ***Risk Agent*** has the ability to assess the risks associated with alternative movement plans based on past performance, current threat conditions, weather forecasts, and political factors.



4. The ***Efficiency Agent*** is responsible for monitoring the compliance of shipments with planned schedules in a reactive mode, and for identifying potential shipment delays or supply-chain disruptions in a proactive mode.



5. The ***Opportunity Agent*** is capable of identifying potential partial conveyance loading based on the ability to algorithmically assess the number of a particular type of conveyance required for a shipment or based on the analysis of cancelled or modified transactions.



6. The ***Closure Agent*** is responsible for determining when a shipment has reached its destination and been delivered, thereby signifying that the movement portion of the transaction has been completed.



7. The ***Load-Planning Agent*** is capable of generating load-plans for ships, aircraft, railcars, and trucks either automatically or in a user-assistance mode, taking into account cargo size and weight, access path, type of conveyance, stability constraints, hazardous material requirements, and cargo spacing tolerances.

Appendix C: Typical Coordination Agents



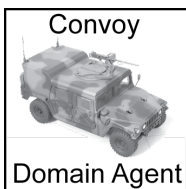
1. The **Conflict Agent** is capable of detecting conflict conditions that may arise among agents and within the transportation network, and identify the likely causes.



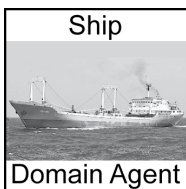
2. The **Collaboration Agent** is responsible for facilitating collaboration by activating agents and alerting the human users to the need for interaction.



3. The **Threat Agent** has the ability to assess threat conditions based on intelligence sources and relate these to individual shipments, as well as the global transportation network by communicating high threat conditions to the *Security Agent*.



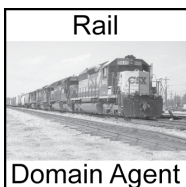
4. The **Convoy Domain Agent** is capable of matching the need for trucks based on load and shipment schedule with the availability of truck convoy transportation from origin to destination (i.e., between the required POE and POD⁷).



5. The **Ship Domain Agent** is capable of matching the need for surface ship transportation, based on cargo list and shipment schedule, with the availability of cargo space on-board vessels moving between the required POE and POD.



6. The **Air Domain Agent** is capable of matching the need for airlift, based on cargo list and air transportation schedule, with the availability of aircraft and aircrews at the designated POE.



7. The **Rail Domain Agent** is capable of matching the need for railcars, based on cargo list and shipment schedule, with the availability of railcars between the nearest railhead and the designated destination (i.e., between the required POE and POD).

⁷ Point of Debarkation (POD).

Appendix D: Typical Governance Agents



1. In the military domain the **Commander's Intent Agent** has the ability to abstract the principal features of a movement plan to a conceptual level for the generation of Commander's Critical Information Requirements (CCIR). In the commercial domain the equivalent objectives are to identify instances when a movement is in serious danger of not meeting stated company objectives.



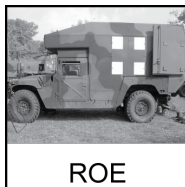
2. The **Performance Agent** has the ability to apply metrics and assess not only the quality of an individual movement plan but also its impact on the overall operational efficiency.



3. The **Priority Agent** is responsible for monitoring the assigned priority of shipments and drawing high priority shipments to the attention of the **Collaboration Agent**, as well as alerting other agents and/or the human user if high priority shipments are subject to delay.



4. The **Security Agent** receives threat condition assessments from the **Threat Agent** and uses these as a basis for determining the appropriate security or force protection (military domain) precautions that should be applied to shipments.



5. The **ROE Agent** (military domain) in collaboration with the designated human user is responsible for maintaining a repository of supply-chain relevant rules of engagement, monitoring the compliance of shipments to these rules, and alerting the designated human user to any ROE violations.