Defence Research and Development Canada | Recherche et développement pour la défense Canada

DEFENCE R&D DÉFENSE

A knowledge-centric view of situation analysis support systems

J. Roy
DRDC Valcartier

Defence R&D Canada – Valcartier
Technical Report
DRDC Valcartier TR 2005-419
January 2007

Canada

A knowledge-centric view of situation analysis support systems

J. Roy
DRDC Valcartier

Defence R&D Canada – Valcartier

Technical Report
DRDC Valcartier TR 2005-419
January 2007

Principal Author

Jean Roy

Defence Scientist

Approved by

Yves Van Chestein

Section Head / IKM, DRDC Valcartier

Approved for release by

Christian Carrier

Chief Scientist, DRDC Valcartier

This document results from S&T work conducted under DRDC projects 11bh (Halifax Class Situation Analysis Support Systems), 11hg (Collaborative Knowledge Exploitation for Maritime Domain Awareness), and 12of (Situation Analysis for the Tactical Army Commander).

Abstract

Situation awareness has emerged as an important concept around dynamic human decision-making in both military and public security environments. Situation analysis is defined as the process that provides and maintains a state of situation awareness for the decision maker(s), and information fusion is a key enabler to meeting the demanding requirements of situation analysis in future command and control (C2) and intelligence support systems. This report presents a new perspective of situation analysis and information fusion (SAIF). The author proposes that developing and adopting a knowledge-centric view of situation analysis should provide a more holistic perspective of this process, leading to the development of better, more adequate SAIF support systems for the operational communities. Along this line of thought, the report provides an overview of many of the main issues around SAIF and knowledge-based systems in order to set up the foundational R&D framework for two projects that are just starting at Defence R&D Canada. Aspects being discussed include C2, intelligence, SAIF, support systems, knowledge, knowledge representation and management, reasoning processes, methods and systems, artificial intelligence, processing/computing/programming paradigms, and expert systems. The effort reported here constitutes some solid basis on which a long-term R&D program in SAIF should be built.

Résumé

L'éveil situationnel a émergé comme concept important de la prise de décision dynamique par des humains dans les milieux militaires et de la sécurité publique. L'analyse de la situation est définie comme étant le processus qui fournit et maintient un état d'éveil situationnel pour le(s) preneur(s) de décisions, et la fusion d'information est un moyen clé pour satisfaire les besoins exigeants de l'analyse de la situation dans les systèmes futurs de soutien au commandement et contrôle (C2) et au renseignement. Ce rapport présente une perspective nouvelle de l'analyse de la situation et de la fusion d'information (ASFI). L'auteur propose que le développement et l'adoption d'une vue de l'analyse de la situation centrée sur la connaissance devrait fournir une perspective plus holistique de ce processus, menant au développement de systèmes de soutien à l'ASFI qui seront meilleurs et plus adéquats pour les communautés opérationnelles. Suivant cette ligne de pensée, le rapport fournit une vue d'ensemble de plusieurs des aspects principaux de l'ASFI et des systèmes à base de connaissances dans le but de créer le cadre de base de R et D pour deux projets qui sont entrepris en ce moment à R et D pour la défense Canada. Les aspects discutés incluent le C2, le renseignement, l'ASFI, les systèmes de soutien, la connaissance, la représentation et la gestion de la connaissance, les processus, méthodes et systèmes de raisonnement, l'intelligence artificielle, les paradigmes de traitement/calcul/programmation, et les systèmes experts. L'effort consigné ici constitue une base solide sur laquelle un programme de R et D à long terme en ASFI devrait être fondé.

This page intentionally left blank.

Executive summary

A knowledge-centric view of situation analysis support systems

Roy, J.; DRDC Valcartier TR 2005-419; Defence R&D Canada – Valcartier; January 2007.

Situation awareness has emerged as an important concept around dynamic human decision-making in both military and public security environments. Situation analysis is defined as the process that provides and maintains a state of situation awareness for the decision maker(s), and information fusion is a key enabler to meeting the demanding requirements of situation analysis in future command and control (C2) and intelligence support systems. This report presents a new perspective of situation analysis and information fusion (SAIF). Mostly based on the fact that *awareness* ultimately has to do with *having knowledge of something*, the author proposes that developing and adopting a knowledge-centric view of situation analysis should provide a more holistic perspective of this process, leading to the development of better, more adequate SAIF support systems for the operational communities.

The report doesn't present “solutions”, or findings and results of completed activities. The intent is more to provide an overview of many of the main issues around SAIF and knowledge-based systems, in order to set up the foundational R&D framework for two projects that are just starting under the Applied Research Program (ARP) at Defence R&D Canada: SATAC (Situation Analysis for the Tactical Army Commander), and CKE-4-MDA (Collaborative Knowledge Exploitation for Maritime Domain Awareness). Aspects being discussed include C2, intelligence, SAIF, support systems, knowledge, knowledge representation and management, reasoning processes, methods and systems, artificial intelligence, processing/computing/programming paradigms, and expert systems.

SAIF has a critical role to play in the C2 and intelligence processes, and it has already received significant attention for military applications. Numerous ongoing projects of the Department of National Defence (DND) and also at Defence R&D Canada have an important SAIF component. However, despite the importance place around SAIF in many DND strategic documents and projects, the current systems and the associated technology being exploited often fail to meet the demanding requirements of the operational decision makers; major science and technology advancements are required to really achieve the full potential of the related enabling technologies and to best serve the operational communities. The effort reported here is one step in this direction; it constitutes some solid basis on which a long-term R&D program should be built.

Although it is somewhat voluminous, this report only provides an initial overview of the many issues arising from the knowledge-centric view of SAIF; one needs to dig deeper into these issues to really embrace this view and develop appropriate support systems meeting the demanding requirements of the operational communities. Hence, R&D activities have been and are still currently being conducted to further investigate knowledge representation concepts, paradigms and techniques, and reasoning processes, methods and systems for use in knowledge-based SAIF support systems. Adopting the knowledge-centric view of SAIF requires that knowledge (i.e., expertise) is eventually acquired from the subject matter experts (SMEs) of the different military and public security application domains. In this regard, knowledge and ontological engineering

techniques have been and are still being investigated at the moment. Knowledge acquisition and validation sessions with SMEs are about to be conducted. Finally, a number of critical research issues in SAIF (e.g., reasoning under uncertainty, exploitation of contextual knowledge, complexity theory) have been briefly discussed in this report; these aspects need to be seriously studied if significant advancements are to be made in this field.

Sommaire

A knowledge-centric view of situation analysis support systems

Roy, J.; DRDC Valcartier TR 2005-419; R et D pour la défense Canada – Valcartier; janvier 2007.

L'éveil situationnel a émergé comme concept important de la prise de décision dynamique par des humains dans les milieux militaires et de la sécurité publique. L'analyse de la situation est définie comme étant le processus qui fournit et maintient un état d'éveil situationnel pour le(s) preneur(s) de décisions, et la fusion d'information est un moyen clé pour satisfaire les besoins exigeants de l'analyse de la situation dans les systèmes futurs de soutien au commandement et contrôle (C2) et au renseignement. Ce rapport présente une perspective nouvelle de l'analyse de la situation et de la fusion d'information (ASFI). Se basant principalement sur le fait que la *conscience* a ultimement un lien avec celui d'*avoir la connaissance de quelque chose*, l'auteur propose que le développement et l'adoption d'une vue de l'analyse de la situation centrée sur la connaissance devrait fournir une perspective plus holistique de ce processus, menant au développement de systèmes de soutien à l'ASFI qui seront meilleurs et plus adéquats pour les communautés opérationnelles.

Le rapport ne présente pas de “solutions”, ou de découvertes et résultats d'activités complétées. L'intention est plutôt de fournir une vue d'ensemble de plusieurs des aspects principaux de l'ASFI et des systèmes à base de connaissances dans le but de créer le cadre de base de R et D pour deux projets qui sont entrepris en ce moment dans le cadre du programme de recherches appliquées (PRA) à R et D pour la défense Canada: ASCAT (Analyse de la situation pour le commandant d'armée tactique), et ECCESDM (Exploitation collaborative de la connaissance pour l'éveil situationnel du domaine maritime). Les aspects discutés incluent le C2, le renseignement, l'ASFI, les systèmes de soutien, la connaissance, la représentation et la gestion de la connaissance, les processus, méthodes et systèmes de raisonnement, l'intelligence artificielle, les paradigmes de traitement/calcul/programmation, et les systèmes experts.

L'ASFI a un rôle critique à jouer dans les processus de C2 et de renseignement, et elle a déjà reçu une attention significative pour les applications militaires. De nombreux projets courants du ministère de la Défense nationale (MDN) et aussi à R et D pour la défense Canada ont une importante composante d'ASFI. Cependant, en dépit de l'importance donnée à l'ASFI dans plusieurs documents et projets stratégiques du MDN, les systèmes actuels et la technologie associée exploitée n'arrivent souvent pas à satisfaire les besoins exigeants des preneurs de décisions opérationnels; des percées majeures en science et technologie sont nécessaires pour atteindre vraiment le plein potentiel des technologies habilitantes associées et pour servir au mieux les communautés opérationnelles. L'effort consigné ici est une étape en ce sens; il constitue une base solide sur laquelle un programme de R&D à long terme devrait être fondé.

Bien qu'il soit plutôt volumineux, ce rapport ne donne qu'une vue d'ensemble initiale de la multitude d'aspects résultants de la vue de l'ASFI centrée sur la connaissance; il est nécessaire d'approfondir ces aspects pour vraiment embrasser cette vue et développer des systèmes de soutien appropriés et répondant aux besoins exigeants des communautés opérationnelles. Pour cette raison, des activités de R et D ont été et sont encore effectuées pour étudier plus à fond les

concepts, paradigmes et techniques de représentation de la connaissance, et les processus, méthodes et systèmes de raisonnement pour leur utilisation dans des systèmes de soutien à l'ASFI basés sur la connaissance. Adopter la vue de l'ASFI centrée sur la connaissance nécessite que la connaissance (c.-à-d., l'expertise) soit éventuellement acquise des experts du domaine (ED) des différents domaines d'application militaires et de la sécurité publique. À ce propos, les techniques d'ingénierie de la connaissance et ontologique ont été et sont encore étudiées en ce moment. Des sessions d'acquisition et de validation de connaissances avec des ED sont sur le point d'être effectuées. Finalement, un certain nombre d'aspects critiques de la recherche en ASFI (p. ex., le raisonnement en incertitude, l'exploitation de la connaissance contextuelle, la théorie de la complexité) ont été discutés brièvement dans ce rapport; ces aspects devront être étudiés sérieusement si des percées significatives doivent être faites dans ce domaine.

Table of contents

List of figures

List of tables

Acknowledgements

The author would like to thank Dr Alain Auger, Defence Scientist at DRDC Valcartier, for some very exciting and productive discussions on knowledge representation. The expertise of Dr Auger in this area has been much appreciated, and it has contributed significantly to the creation of Fig. 28: *The main elements of knowledge representation*.

1. Introduction

It is generally accepted that Command and Control (C2) is composed of a number of dynamic and cyclic perceptual, procedural and cognitive activities, achieved either by humans, computer systems or both. At a very high level, these activities can be summarized as the perception of the environment, the appraisal of the situation, the decision making about a course of action, and the implementation of the chosen plan. The Observe-Orient-Decide-Act (OODA) loop is often used to describe the C2 process at that level. The essence of C2 in military and public security operations is thus people making timely decisions in the face of uncertainty, and acting on them. At the heart of the C2 process is the provision of decision quality information to the decision maker, thereby enabling timely understanding of the dynamic situation. This is a timeless requirement, which has been immeasurably complicated by the overwhelming and increasing volume of raw data and information available in the current age.

Intelligence refers to a special kind of knowledge necessary to accomplish a mission, i.e., the kind of strategic knowledge that reveals critical threats and opportunities that may jeopardize or assure mission accomplishment [Waltz, 2003]. It is knowledge and foreknowledge of the world around us, the prelude to decision and action. In this rapid changing world, the expectations regarding those in the intelligence discipline are high. The consumers of intelligence all expect accurate and timely information about their areas of interest and the threats to their security. As for C2, the process that delivers strategic and operational intelligence products is generally depicted in cyclic form [Waltz, 2003], with distinct constituents for obtaining, assembling and evaluating information, converting it into intelligence, and disseminating it [McIntyre, Gauvin, Waruszynski, 2003]. The processing phase of the intelligence cycle involves the collation, evaluation, analysis, integration and assessment of the gathered information. The organized information base is processed using estimation and inferential (reasoning) techniques that combine all-source data in an attempt to answer the requestor's questions. The data is analyzed (broken into components and studied) and solutions are synthesized (constructed from the accumulating evidence).

There are similarities between the C2 OODA loop and the intelligence cycle. One such similarity has to do with the notion of situation awareness (SAW) that has emerged as an important concept around dynamic human decision-making in military and public security environments. The two loops hinge on the fulfillment of a broad function, i.e., that commanders arrive at a consistent understanding of the battlespace, arising through battlespace (situation) awareness. Actually, SAW is a general concept that has been shown to be of interest in a very large number of settings.

Given this wide interest for SAW, the author has previously proposed another concept, situation analysis (SA), defined as a process that provides and maintains a state of situation awareness for the decision maker(s) [Roy, 2001]. A key enabler to meeting the demanding requirements of situation analysis in future C2 and intelligence systems (i.e., in achieving high-quality situation awareness for optimal decision making) is data/information fusion. An initial lexicon defined data fusion as a process dealing with the association, correlation, and combination of data and information from single and multiple sources to achieve refined position and identity estimates, and complete and timely assessments of situations and threats as well as their significance [White, 1987]. This definition has evolved over the years, and multiple variants have been proposed. Recently, [Lambert, 2001] defined information fusion as the process of utilizing one or more information sources over time to assemble a representation of aspects of interest in an environment. Among the many reasons for interest in this technology, there is that data/information fusion:

- provides extended spatial and temporal coverage, increased confidence, reduced ambiguity, improved entity detection, etc.,
- allows for the management of large volumes of information, and the correlation of seemingly unrelated, overlooked, or deceptive information to present a coherent representation of an evolving situation to a decision maker, and,
- enables the commander to cope with the complexity and tempo of operations in the modern dynamic operational theatres.

Clearly situation analysis and information fusion (SAIF) has a critical role to play, and it has already received significant attention for military applications. Numerous ongoing projects of the Department of National Defence (DND) have an important SAIF component. Examples, just to name a few, are:

- Project No. 00000276: Land Force Intelligence, Surveillance, Target Acquisition and Reconnaissance (LF ISTAR)
- Project No. 00000806: Marine Security Operations Centres (MSOC)
- Project No. 00000624: Joint Information and Intelligence Fusion Capability (JIIFC)

However, despite the importance place around SAIF in many DND strategic documents and projects, the current supporting systems and the associated technology being exploited often fail to meet the demanding requirements of the operational decision makers, and major Science and Technology (S&T) advancements are required to really achieve the full potential of the related enabling technologies and to best serve the operational communities. In this line of thought, many R&D projects ongoing under the Technology Demonstration Program (TDP) at Defence R&D Canada (DRDC) have an important SAIF component as well:

- Joint Command Decision Support for the 21st Century (JCDS 21)
- Innovative Naval Combat Management Decision Support (INCOMMANDS)

SAIF is certainly expected to play a crucial role in the next generation of support systems for aiding decision makers in military and public security operations.

Taking into account the context described above, the purpose of this report is to present a new perspective of SAIF. The author proposes that developing and adopting a knowledge-centric view of situation analysis should provide a more holistic perspective of this process, leading to the development of better, more adequate SAIF support systems for the operational communities. This is based on a number of notions:

- *Awareness* ultimately has to do with *having knowledge of something*.
- Intelligent agents need knowledge about the world in order to reach good decisions.
- Not all of the situation elements and relationships of interest to a decision maker are directly observable; those aspects of interest that cannot be observed must be inferred, i.e., derived as a conclusion from facts or premises, or by reasoning from evidence.

- There's been a shift of paradigm in Artificial Intelligence (AI) from the pursuit of powerful search and reasoning methods toward a recognition of the role of special case knowledge, i.e., from a technique-oriented theory of intelligence to a knowledge-oriented theory of intelligence [Stefik, 1995]. The fundamental problem of understanding intelligence is not the identification of a few powerful techniques, but rather the question of how to represent large amounts of knowledge in a fashion that permits their effective use and interaction [Stefik, 1995].

The report doesn't present “solutions”, or findings and results of completed activities. The intent is more to provide an overview of many of the main issues around SAIF and knowledge-based systems, in order to set up the foundational R&D framework for two DRDC projects that are just starting under the Applied Research Program (ARP):

- SATAC (Situation Analysis for the Tactical Army Commander), and,
- CKE-4-MDA (Collaborative Knowledge Exploitation for Maritime Domain Awareness), formerly known as AKAMIA (Advanced Knowledge Acquisition for Maritime Information Awareness).

As alluded above, SAIF is not something that happens in a vacuum, and it should not be decoupled from the decision making process. This is the reason why it is discussed in the context of command and control and intelligence in this report. Moreover, although human qualities such as initiative, creativity, and the notions of responsibility and accountability remain essential to C2 and intelligence, the support of the technology is clearly required to cope with the demanding characteristics inherent to these domains (e.g., time-critical conditions, saturation, etc.), in order to complement human capabilities and address its limitations. Hence, SAIF is also discussed here from the perspective of computer-based support systems.

SAIF draws together concepts from a wide range of diverse fields: psychology, human factors, cognitive engineering, knowledge representation, ontology, knowledge management, knowledge engineering, artificial intelligence, reasoning/inference, mathematics, logic, expert systems, signal processing, computer science, etc. Many of these aspects are discussed in this report, at various levels of depth.

The report is organized as follows. To set the appropriate background for SAIF, Section 2 provides an overview of command and control (C2) and intelligence. C2 is presented mostly from the point of view of Alberts and Hayes and the U.S. Department of Defense (DoD) Command and Control Research Program (CCRP). Intelligence and the intelligence cycle are both introduced, and a link is then made between C2 (or the OODA loop) and the intelligence cycle.

Section 3 briefly introduces the notions of situation analysis and awareness, and Section 4 then discusses situation analysis and knowledge exploitation in the perspective of support systems. The need for technological support and the ideal support system are first discussed, and then the components of a generic support system are presented. Beginning with a short discussion on *all information* versus *the right information*, Section 5 then provides a definition of data and information fusion and discusses many related concepts. The data fusion model maintained by the Joint Directors of Laboratories’ Data and Information Fusion Group (JDL DIFG) is presented.

As previously mentioned, not all of the situation elements of interest to a given decision maker are directly observable with the typical data and information sources currently available. Those aspects of interest that cannot be observed must be inferred. This is an essential aspect of situation analysis that is discussed in Section 6, and that will need a lot more attention in the foreseeing future. The concept of a “processing node”, both for information fusion and for inference, is presented.

Section 7 then introduces the knowledge-centric view of situation analysis support systems. It discusses the main relationships between situation analysis and awareness on one hand, and information fusion and knowledge-based systems on the other hand. At the heart of the knowledge-centric view is knowledge. This being said, putting knowledge into computers raises many foundational questions: What is knowledge? Where does it come from? How is it created? How is it held by computers? Section 8 initiates the discussions on such issues. Definitions of the term *knowledge* are first provided, followed with discussions on the distinction between the notions of knowledge, information, and data (KID), and also on the cognitive hierarchy. Various categories of knowledge are reviewed, and examples of typical sources of knowledge are provided. Given their importance for knowledge-based situation analysis support systems, expertise (a specialized type of knowledge) and the notion of “an expert” are discussed.

For a number of reasons, knowledge management (KM) has emerged as an important discipline in the military context. This is the topic of Section 9 that first discusses *knowledge as an object* versus *knowledge as a process.* A model of knowledge creation and transfer is presented, built around the notions of externalization, combination, internalization and socialization. Then, some definitions of KM are provided, and the KM cycle and its enabling disciplines are presented.

One cannot put the world in a computer, so all computer-based reasoning mechanisms must operate on representations of facts, rather than on the facts themselves. In this regard, the object of knowledge representation (KR) is to express knowledge in computer-tractable form, such that it can be exploited. This is the theme of Section 10, where the main elements of knowledge representation are briefly introduced. This is followed with a short discussion on the reasoning processes, methods and systems in Section 11. Knowledge representation and automated reasoning are key concepts of artificial intelligence (AI), the interdisciplinary field that is the subject of Section 12. Intelligent behaviour, testing for intelligence, and *artificial* versus *natural* intelligence are all discussed in this section. As AI is the branch of computer science dealing primarily with symbolic, nonalgorithmic methods of problem solving, *numerical* versus *symbolic* and *algorithmic* versus *nonalgorithmic* processing are discussed in Section 13 that's about processing, computing and programming paradigms. Conventional versus AI computing and programming paradigms and languages are also reviewed in this section.

Expert systems are a branch of AI that makes extensive use of specialized knowledge to solve problems at the level of a human expert. They are the paradigmatic application of AI techniques to hard problems, and they form the subject of Section 14. Among other things, this section discusses the distinction between expert and knowledge-based systems, knowledge representation in expert systems, elements of a rule-based expert system, and the types and desirable characteristics of expert systems. A comparison is made between *conventional* and *expert* systems, and the benefits and limitations of knowledge-based expert systems are presented, along with suitable application domains and success factors for expert systems.

Finally, some critical research issues in situation analysis and information fusion are briefly discussed in Section 15, and concluding remarks are provided in Section 16.

2. Command and Control and Intelligence

2.1 Command and Control

Understanding command and control (C2) is no longer an option; it is a requirement [Alberts, Hayes, 2006]. We need to understand C2 thoroughly if we want to make significant progress on defence and public security transformation, or succeed in 21st century operations.

Clearly, the challenges of the 21st century missions have increased significantly. Today's missions differ from traditional missions, as they are simultaneously more complex and more dynamic, requiring the collective capabilities and efforts of many organizations in order to succeed [Alberts, Hayes, 2006]. This requirement for assembling a diverse set of capabilities and organizations into an effective partnership is accompanied by shrinking windows of response opportunity.

Simply stated, C2 is the common military term for the management of personnel and resources [Alberts, Hayes, 2003]. This being said, a consensus has not yet been attained over a single, complete and precise definition of the term C2. In this regard, the official definition provided by the Department of Defense (DoD) in the United States is often quoted. According to the U.S. DoD Dictionary of Military and Associated Terms [DoD, 2006], C2 is defined as:

> *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.*

Beyond its definition, command and control is not an end in itself, but it is a means toward creating value (e.g., the accomplishment of a mission). Specifically, C2 is about focusing the efforts of a number of entities (individuals and organizations) and resources, including information, toward the achievement of some task, objective, or goal [Alberts, Hayes, 2006]. How C2 (or management) is or may have been done in industry and military organizations should not be equated with why C2 (or management) is needed or what functions need to be successfully performed to create value.

Command and control is a relatively recent term that for millennia was referred to as simply command [Alberts, Hayes, 2003]. Command concepts both predate and have evolved separately from politics and industrial management. This is because warfare is qualitatively different from the management of other human enterprises by virtue of its time criticality and the high cost of error. Both of these characteristics of warfare have shaped thinking about C2.

Although the purpose of C2 has remained unchanged since the earliest military forces engaged one another, the way we have thought about it, and also the means by which the functions of C2 have been accomplished, have changed significantly over the course of history [Alberts, Hayes, 2006]. The models discussed in the next two sections reflect this evolution.

2.1.1 A View of the Command and Control Process: The OODA Loop

[Boyd, 1987] introduced the Observe, Orient, Decide, and Act (OODA) loop in order to support the analysis of pilot decision making at a tactical level [Alberts et al, 2001]. The idea, illustrated in Fig. 1, is that decisions begin by observing the physical domain. The observations are then placed in the context of other information and prior knowledge (so they become useful information) in order to orient the individual, which (in turn) allows this individual to decide what is to be done and act accordingly. The concept has proved to have considerable intuitive appeal and has been used for decades as the basis of both analysis and training. The phrase "turning inside the enemy's OODA loop", while originating in air-to-air combat, has become the shorthand way of understanding that the speed of the C2 process can provide advantage in combat situations.

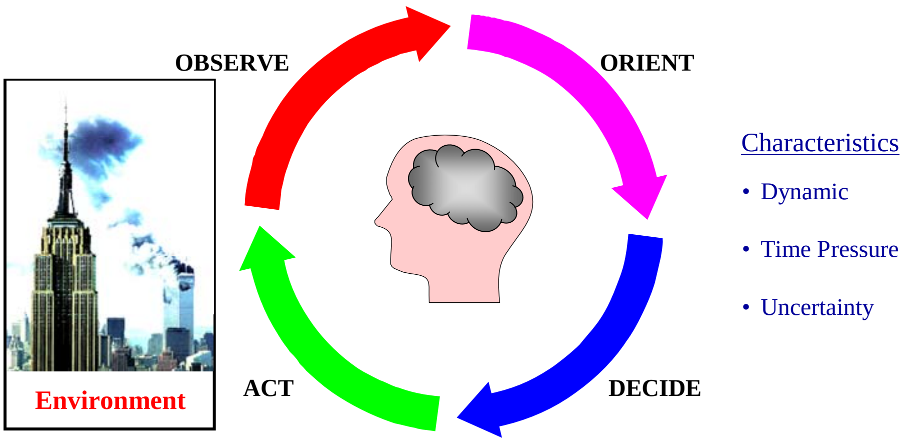

Figure 1: The OODA loop model [Boyd, 1987]

In the language of [Alberts et al, 2001], the act of observation must begin in the physical domain, may pass through some fusion with other observations, and is brought to the individual's attention through the information domain. The process of orientation occurs in the cognitive domain as the information content of the observations is internalized and placed in the context of the individual's prior knowledge, experience, and training. This is seen as providing the basis for a decision – also a cognitive activity. Finally, the decision itself must pass through the information domain (e.g., the controls of an aircraft, the directives of a commander) in order to become the basis for action.

The OODA loop has proven seductively robust and has been applied not only to pilot's activities in air-to-air combat, but also to organizational behaviour at all levels. However, in the view of [Alberts et al, 2001], this is an error. The OODA loop both oversimplifies the command and control process in ways that confound analysis and also reifies military organizations – implying that they have a single mind and make a single, coordinated decision across echelons and functions. [Alberts et al, 2001] believe that the OODA loop is outdated because it fails to differentiate crucial elements that must be considered in information age analyses. Moreover, the OODA loop greatly oversimplifies the joint hierarchical model underlying military operations.

2.1.2 An Information Age View of the Command and Control Process

Figure 2 provides an information age view of the traditional C2 process as it has been understood for several decades. However, it uses much richer constructs than those in the OODA loop [Alberts et al, 2001]. In contrast to the logic in the simpler OODA loop construct, which sees the output of the cognitive processes as a decision, the information age C2 process is understood to generate a richer product – command intent. This choice of language has two important, direct implications: 1) the product is much richer, and 2) more than one individual is involved.

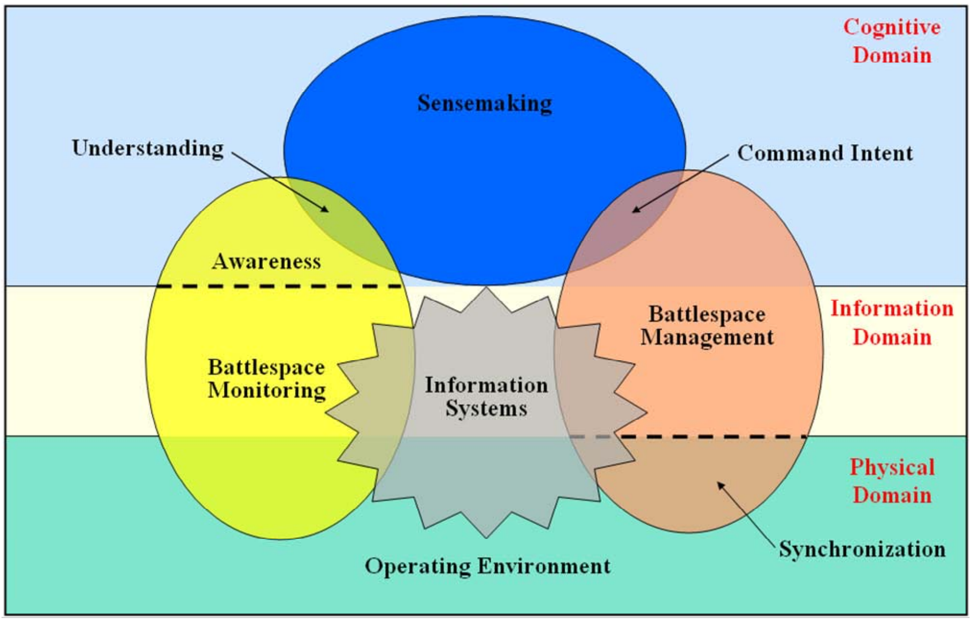

Figure 2: An information age view of the C2 process [Alberts et al, 2001]

As illustrated in Fig. 2, given that the term command and control encompasses as much as it does, its elements span all of the three domains of warfare (i.e., physical, information, cognitive) that were introduced in the previous sub-section. C2 sensors, systems, platforms, and facilities exist in the physical domain. The information collected, posted, pulled, displayed, processed, and stored exists in the information domain. The perceptions and understanding of what this information states and means exist in the cognitive domain. Also in the cognitive domain are the mental models, preconceptions, biases, and values that serve to influence how information is interpreted and understood, as well as the nature of the responses that may be considered. Finally, C2 processes and the interactions between and among individuals and entities that fundamentally define organization and doctrine exist in the social domain (not shown in Fig. 2).

More recently, [Alberts, Hayes, 2006] identified the following functions associated with the command and control (or management) of a given undertaking:

- Establishing intent (the goal or objective)
- Determining roles, responsibilities, and relationships
- Establishing rules and constraints (schedules, etc.)
- Monitoring and assessing the situation and progress

- Inspiring, motivating, and engendering trust
- Training and education
- Provisioning

These C2 functions are applicable not only to military endeavours but also to civil-military and indeed to civilian and industrial enterprises. Each of these functions can be seen in the context of a particular time horizon.

In the continuity of their effort to better understand C2, [Alberts, Hayes, 2006] have also proposed another conceptual model of command and control, shown in Fig. 3.

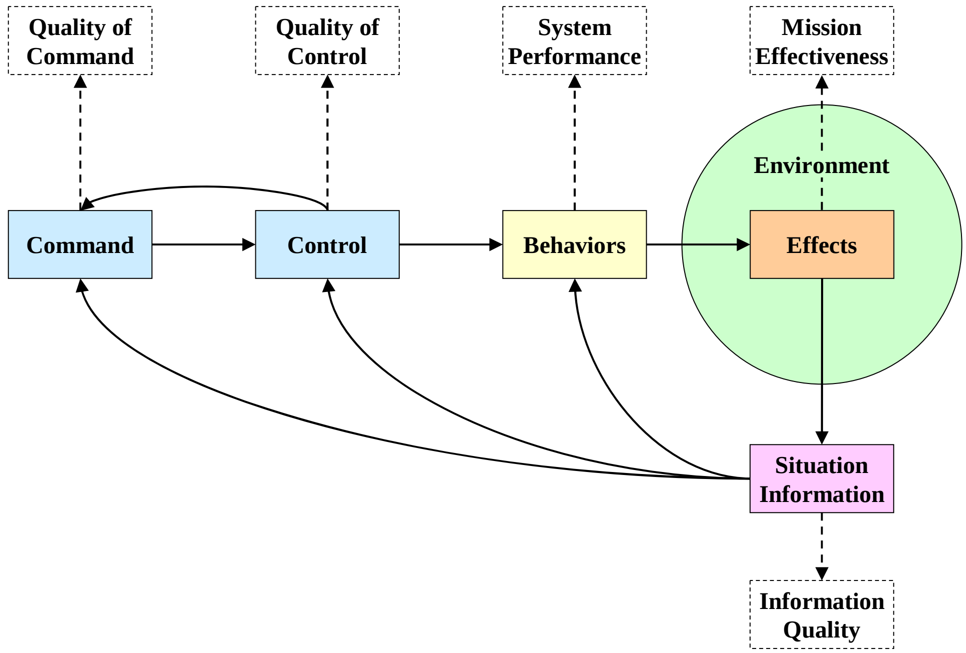

Figure 3: Command and control conceptual model [Alberts and Hayes, 2006]

According to the U.S. DoD Dictionary of Military and Associated Terms [DoD, 2006], *command* is defined as:

> *The authority that a commander in the Armed Forces lawfully exercises over subordinates by virtue of rank or assignment. Command includes the authority and responsibility for effectively using available resources and for planning the employment of, organizing, directing, coordinating, and controlling military forces for the accomplishment of assigned missions. It also includes responsibility for health, welfare, morale, and discipline of assigned personnel.*

This definition subsumes control as a part of command. However, many have tried to draw a distinction between command and control [Alberts, Hayes, 2003]. Distinctions that have been drawn include one between art (command) and science (control), and one between the commander (command) and staff (control). Hence, rather than treating C2 as a single concept, [Alberts, Hayes, 2006] have chosen to separate command from control to maintain the greatest degree of flexibility. This enables them to examine each concept on its own and combine different approaches to each in ways that have not been considered before. Thus, they start constructing their conceptual model (Fig. 3) with two boxes, one representing the concept of command, and the other representing the concept of control. Together, these two boxes define the C2 space.

Prior to the commencement of an operation, intent (a command function) needs to be established [Alberts, Hayes, 2006]. This intent can consist of merely recognizing that there is a situation that needs to be dealt with or a problem to be solved. It does not require that a solution or an approach be developed. Roles, responsibilities, and relationships may be predetermined or they may be established or modified to suit the circumstances (intent and the situation). The establishment of a role determines whether or not the entity is considered part of the team or part of the environment. Likewise, rules and constraints and resource allocations can be predetermined or tailored to the situation.

Once an operation begins (and this dates from the establishment of intent, not from the commencement of a response, where response can include pre-emptive action), intent can change, as can roles, responsibilities, allocations of resources, and the like [Alberts, Hayes, 2006]. All of these changes to the set of initial conditions, with the exception of a change to intent, should be considered control functions. Changing intent is a command function. The ability to make timely and appropriate changes is directly related to the agility of the specific instantiation of a C2 approach. Given the complexity of the 21st century security environment and the missions that 21st century militaries are and will be called upon to accomplish, C2 agility is perhaps the most important attribute of a C2 approach. The establishment and communication of the initial set of conditions, the continuing assessment of the situation, and changes to intent are functions of command. The ability to exercise command (the accomplishment of the functions associated with command) is affected or influenced by, among other things, the quality of information available.

The function of control is to determine whether current and/or planned efforts are on track [Alberts, Hayes, 2006]. If adjustments are required, the function of control is to make these adjustments if they are within the guidelines established by command. The essence of control is to keep the values of specific elements of the operating environment within the bounds established by command, primarily in the form of intent.

Behaviours include [Alberts, Hayes, 2006]:

- those actions and interactions among the individuals and organizations that accomplish the functions associated with C2 (e.g., establishing intent, conveying intent),
- those that are associated with understanding or making sense of the situation and how to respond, and,
- those that are associated with the response (that is, with creating the desired effects such as manoeuvre and engagement).

The first two sets of behaviours constitute C2. The second set of behaviours is a subset of C2 called sensemaking, and the third set of behaviours can be referred to as actions or execution. All are functions of an enterprise (organization or endeavour). Both the objective of sensemaking and execution and how they are accomplished are determined by command and control.

Sensemaking consists of a set of activities or processes in the cognitive and social domains that begins on the edge of the information domain with the perception of available information and ends prior to taking action(s) that are meant to create effects in any or all of the domains. Examples are [Alberts, Hayes, 2006]:

- The employment of kinetic weapons with direct effects in the physical domain and indirect effects in the other domains.
- The employment of psychological or information operations designed to create direct effects in the cognitive and information domains with indirect effects in the physical domain.

The actions involved in execution may take place in any of the domains with direct and indirect effects in multiple domains. The nature of the effects created by a particular action are a function of 1) the action itself, 2) when and under what conditions the action is taken, 3) the quality of the execution, and 4) other related actions. The selection of what actions to take and when to take them is part of the sensemaking process.

The operating environment includes everything outside of the C2 processes and the systems that support those [Alberts et al, 2001]. The physical environment (terrain, weather, etc.) is one key dimension. Adversary forces form another. Own forces, to the extent that they are not part of C2 processes, are also in the environment. They represent the most controllable factors in the environment, but even they are imperfectly controllable due to the fog and friction of war. Other, neutral forces may also be present in the portion of the operating environment of interest. Their potential involvement or interference must also be considered. The operating environment also includes a host of political, social, and economic factors and actors, ranging from refugee populations to the infrastructure (communications, transportation, etc.) in the area.

[Alberts, Hayes, 2006] state that the context of C2 can greatly vary. The nature of the tasks at hand differ widely, ranging from the creation or transformation of an enterprise at the strategic level, to employing the enterprise in a major undertaking at the operational level, to the completion of a specific task at the tactical level. As the nature of the task differs, so does the nature of the resources involved. These can range from something that can be accomplished with organic assets to something that requires putting together a large heterogeneous coalition with resources of many types.

In the past, much of the discussions about C2 were often focused on a single commander, the one in charge. In fact, command and control in modern warfare is a distributed responsibility. Actually, the C2 conceptual model depicted in Fig. 3 is elemental or fractal [Alberts, Hayes, 2006]. An enterprise of the complexity necessary to undertake military and civil-military missions will have many concurrent, nested, and even overlapping instances of this elemental model, each one of (or collection of) which may exhibit different C2 approaches. At the enterprise level, the functions associated with command will determine the number and nature of these fractals and the relationships among them. Thus, if we consider Fig. 3 to be a view at the enterprise level, then there will be a great many “little Fig. 3s” contained in the enterprise view of the behaviours box, or for that matter, the boxes for command and for control. Command at one level determines the conditions under which fractals that are within their purview operate. There will be cases of sovereign fractals in which the fractals are not nested but have peer-to-peer and/or overlapping relationships. In these cases, the functions associated with C2 are achieved in a manner different from that of traditionally nested fractals.

The conceptual model of Fig. 3 consists of two kinds of concepts: 1) functional or process concepts, and 2) concepts related to value. A generic process view of the conceptual model is depicted in Fig. 4.

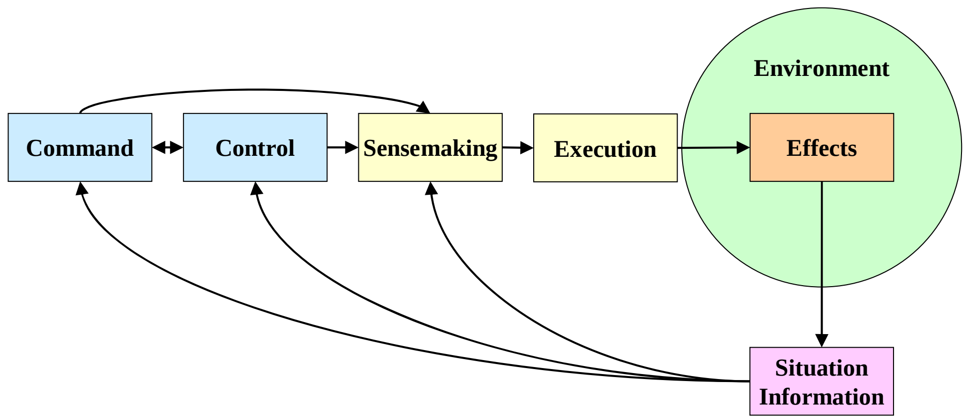

Figure 4: C2 conceptual model - A process view [Alberts and Hayes, 2006]

The process view organizes functions and processes, whether past, current, or future, into a small number of conceptual bins. The bin “situation information” represents the host of information-related assets and processes that sense, collect, process, protect, disseminate, and display information. The product of these processes (information) provides data about the environment, including the effects of interest. This information is used as an input to all of the other process concepts.

2.2 Intelligence

Intelligence refers to a special kind of knowledge necessary to accomplish a mission, i.e., the kind of strategic knowledge that reveals critical threats and opportunities that may jeopardize or assure mission accomplishment [Waltz, 2003]. In this rapid changing world, the expectations required of those in the intelligence discipline are high:

- Knowledge of the hidden.
- Foreknowledge of the unpredictable.

Intelligence often reveals hidden secrets or conveys a deep understanding that is covered by complexity, deliberate denial, our outright deception [Waltz, 2003]. It is knowledge and foreknowledge of the world around us, the prelude to decision and action. These classical components of intelligence, i.e., knowledge and foreknowledge, provide the insight and warning that leaders need for decision making to provide security.

The consumers of intelligence all expect accurate and timely information about their areas of interest and threats to their security [Waltz, 2003]. They want strategic analyses, indications and warnings, and tactical details. From a torrent of data, real-world intelligence produces a steady stream of reliable and actionable knowledge.

2.2.1 The Intelligence Cycle

Intelligence has always involved the management (acquisition, analysis, synthesis, and delivery) of knowledge [Waltz, 2003]. Real-world intelligence is not a puzzle of connecting dots; it is the hard daily work of planning operations, focusing the collection of data, and then processing the collected data for deep analysis to produce a flow of knowledge for dissemination to a wide range of consumers.

The process that delivers strategic and operational intelligence products is generally depicted in cyclic form [Waltz, 2003], with distinct phases for obtaining, assembling and evaluating information, converting it into intelligence, and disseminating it [McIntyre, Gauvin, Waruszynski, 2003]. This is illustrated in Fig. 5. The cycle begins with the need for knowledge by policy or decision makers (consumers) and concludes with the delivery of that knowledge. The need may be a standing requirement, a special request, or an urgent necessity in time of crisis. In every case, the need is the basis for a logical process to deliver the knowledge to the requestor [Waltz, 2003].

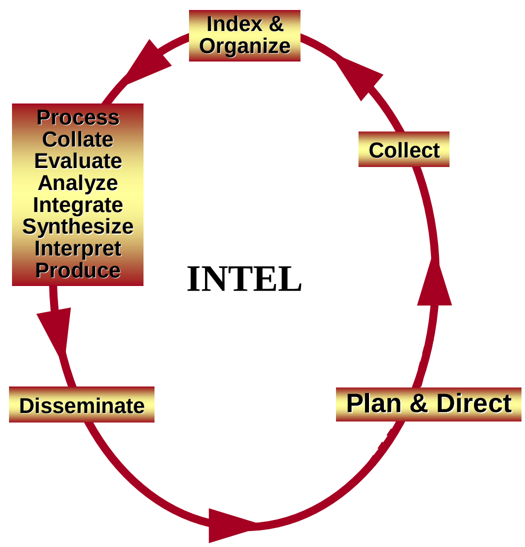

Figure 5: The intelligence cycle

The first phase of the intelligence cycle is when commanders determine the requirements and communicate them to the staff [McIntyre, Gauvin, Waruszynski, 2003]. The process begins as policy and decision makers define, at a high level of abstraction, the knowledge that is required to make decisions [Waltz, 2003]. The requests are parsed into information required, then to data that must be collected to estimate or infer the required answers. Data requirements are used to establish a plan of collection, which details the elements of data needed and the targets (people, places, and things) from which the data may be obtained. Following the plan, human and technical sources of data are tasked to collect the required raw data. Reconnaissance and surveillance data is gathered by sources and agencies. The collected data is pre-processed (e.g., machine translation, foreign language translation, or decryption), indexed, and organized in an information base. Progress on meeting the requirements of the collection plan is monitored and the tasking may be refined on the basis of received data.

The processing phase involves the collation, evaluation, analysis, integration and assessment of the gathered information [McIntyre, Gauvin, Waruszynski, 2003]. The organized information base is processed using estimation and inferential (reasoning) techniques that combine all-source data in an attempt to answer the requestor's questions [Waltz, 2003]. The data is analyzed (broken into components and studied) and solutions are synthesized (constructed from the accumulating evidence). The topics or subjects (intelligence targets) of study are modeled, and requests for additional collection and processing may be made to acquire sufficient data and achieve a sufficient level of understanding (or confidence to make a judgment) to answer the consumer's questions. This phase is the conversion of information into intelligence [McIntyre, Gauvin, Waruszynski, 2003].

In the final phase, dissemination, intelligence is distributed to those who require it [McIntyre, Gauvin, Waruszynski, 2003]. Finished intelligence is disseminated to consumers in a variety of format, providing answers to queries and estimates of accuracy of the product delivered [Waltz, 2003]. Though introduced here in the classic form of a cycle, in reality the process operates as a continuum of actions with many more feedback (and feedforward) paths that require collaboration between consumers, collectors, and analysts.

2.3 Linking the OODA Loop and the Intelligence Cycle

Variations of Fig. 6 are often used to illustrate the link and similarities between the OODA loop (used at a high level to describe the C2 process during military or public security operations) and the intelligence cycle. This figure is an attempt to summarize in some sort the C4ISR world. Commanders exercise C2 by synchronizing military actions in time, space, and purpose to achieve unity of effort within a military force under two main constraints – uncertainty and time – that dominate the environment in which military decisions are made.

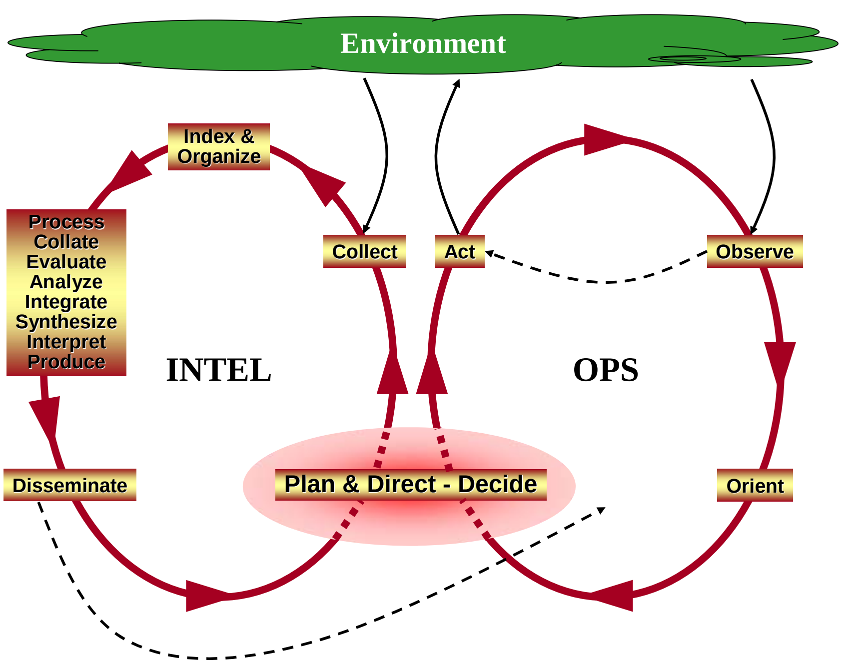

Figure 6: Linking the OODA loop and the intelligence cycle

The OODA loop and the intelligence cycle are two models for the information and decision-making processes. The two loops hinge on the fulfillment of two broad functions: first, that all commanders within a force arrive at a shared and consistent understanding of the battlespace arising through *battlespace (situation) awareness*; and, second, that *unity of effort* is achieved *(decision making)* throughout a joint and combined force through commonly held intent. From the perspective of situation analysis, data and information fusion, and reasoning or inference, these two cycles present many similar characteristics.

3. Situation Analysis and Awareness

This section briefly introduces the notions of situation analysis and awareness that play a critical role in C4ISR.

3.1 Awareness

Situation awareness (SAW) has emerged as an important concept around dynamic human decision making, especially in complex military and national security environments. As illustrated in Fig. 7, the idea of awareness has to do with having knowledge of something [Roy, 2001]; this is also true of the synonyms aware, cognizant, and conscious [Merriam-Webster, 2003].

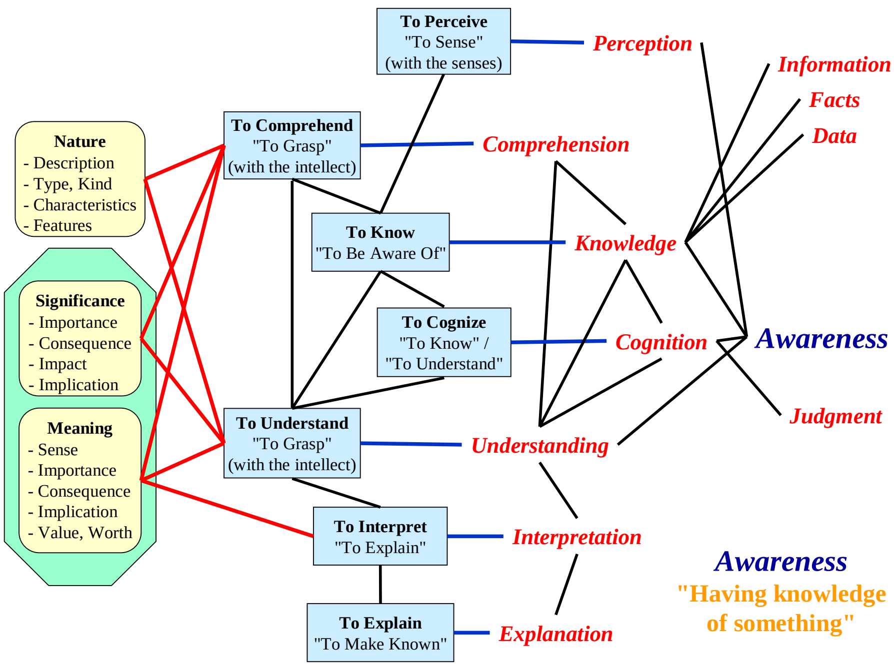

Figure 7: Awareness, or "having knowledge of something" [Roy, 2001]

[Endsley, 1995] has proposed a general definition of situation awareness that has been found to be applicable across a wide variety of domains. She describes SAW as the perception of the elements in the environment within a volume of time and space, the comprehension of their meaning and the projection of their status in the near future. This is illustrated in Fig 8.

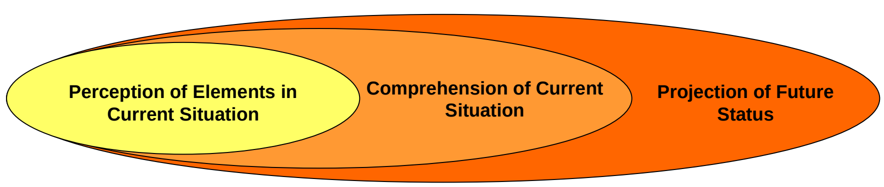

Figure 8: Situation awareness [Endsley, 1995]

The perception of cues is fundamental. Without a basic perception of important information, the odds of forming an incorrect assessment of the situation increase dramatically [Endsley, Garland, 2000].

Endsley also states that SAW as a construct goes beyond mere perception. It also encompasses how people combine, interpret, store, and retain information. Thus, it includes more than just perceiving or attending to information; it includes the integration of multiple pieces of information and a determination of their relevance to the person's goals. Such a comprehension of a situation demands that the problem of *meaning* be tackled head-on. A person with situation comprehension has been able to derive operationally relevant meaning and significance from the data perceived.

At the highest level of SAW, the ability to forecast future situation events and dynamics marks decision makers who have the highest level of understanding of the situation. This ability to project from current events and dynamics to anticipate future events (and their implications) allows for timely decision making.

3.2 Situation Analysis

In her model, Endsley presents situation awareness as a stage separate from decision making (DM) and action. SAW is described as the decision maker's internal, mental model of the state of the environment. Based on that representation, the decision maker can decide what to do about the situation and carry out any necessary actions. There is thus a strong link between SAW and the DM processes. SAW is represented as the main precursor to decision making. This is illustrated in Fig. 9, built around Boyd's OODA loop.

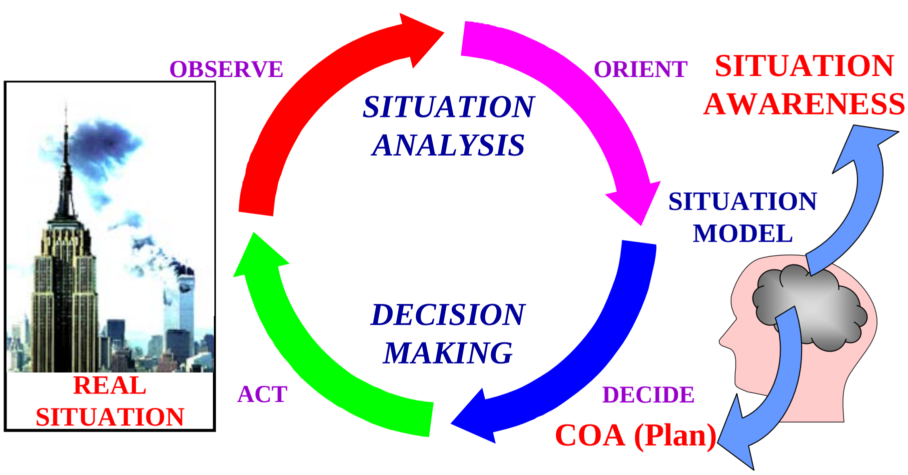

Figure 9: Situation analysis and decision making [Roy, 2001]

In this perspective, [Roy, 2001] has proposed another concept, Situation Analysis (SA), in an attempt to merge into a synthesis the main notions put forward by well-established data fusion and situation awareness models. He defines situation analysis as a process, the examination of a situation, its elements, and their relations, which provides and maintains a product, i.e., a state of situation awareness, for the decision maker.

As shown in Fig. 9, the SA process encapsulates that part of the overall decision-making cycle that is concerned with understanding the world. There is a real situation in the environment, and the SA process will create and maintain a mental representation of it, the situation model, in the head of the decision maker(s).

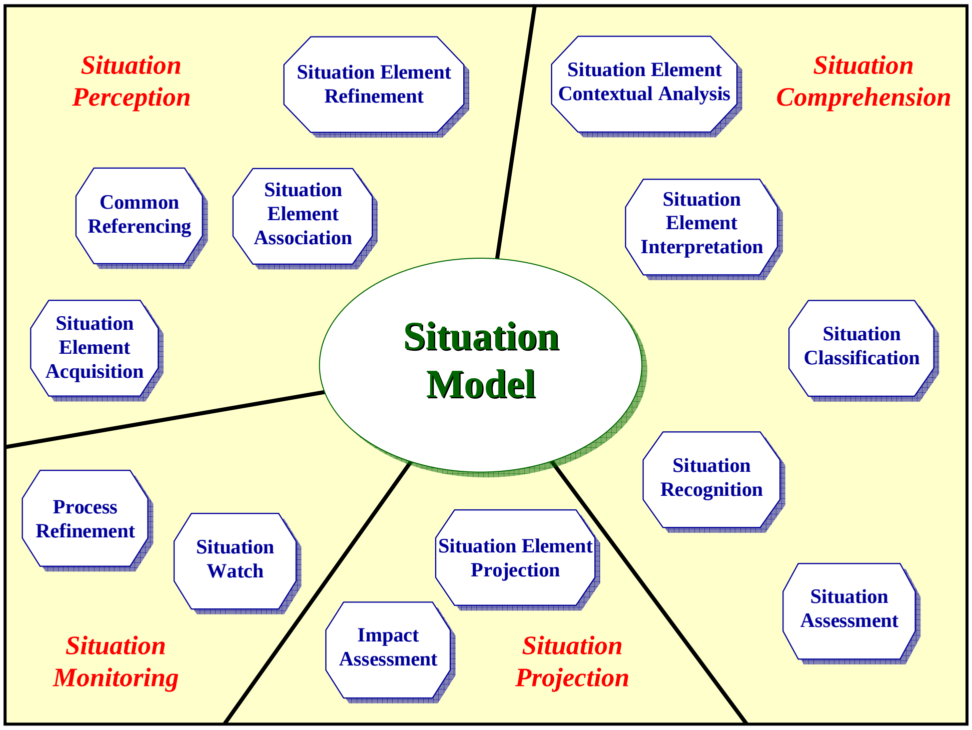

Figure 10: The situation analysis process (high-level view) [Roy, 2001]

At the highest level, making a strong parallel with Endsley's work, the SA process can be decomposed into four sub-processes: situation perception, comprehension, projection and monitoring (note that we are talking processes here, not states). This is shown on Fig. 10.

Situation perception has to do with the “acquisition” of the situation through data/information collection with various sensors and other sources. Situation comprehension is about further developing one's knowledge of the situation with respect to both its nature, i.e., the inherent character or basic constitution of the situation, and its significance or meaning, i.e., the importance, the consequence, the impact and/or the implication of the situation. This sub-process must be able to grasp the nature of the situation, and to derive operationally relevant meaning and significance from the results of situation perception. Situation projection must produce an estimate of future possibilities for situation elements, based on current trends. Finally, situation monitoring has to do with watching, observing, or checking the evolution of the situation in order to ultimately keep track of, regulate or control the operation of the SA process.

Figure 11 is a detailed functional description of the SA process. From a data-driven perspective, it entails integrating and interpreting the whole spectrum of source data and information, ranging from radar returns to political factors. The SA process thus encompasses a vast range of activities, from the detailed signal processing associated with target acquisition and tracking, to intelligence interpretation. Simply put, the process must provide answers to a great number of questions:

What? Who? How many? How big? Where? What structure? When? What is it doing? Why? Build up? What could it do? How soon? What is outstanding? What has changed? Delta from expectations? What is going wrong? and many others.

The SA process thus consists of numerous dependent and independent sub-processes, or capabilities, at multiple levels of abstraction. Every sub-process can itself be further decomposed hierarchically into multiple sub-processes. These SA capabilities must be integrated and interleaved into an overall processing flow.

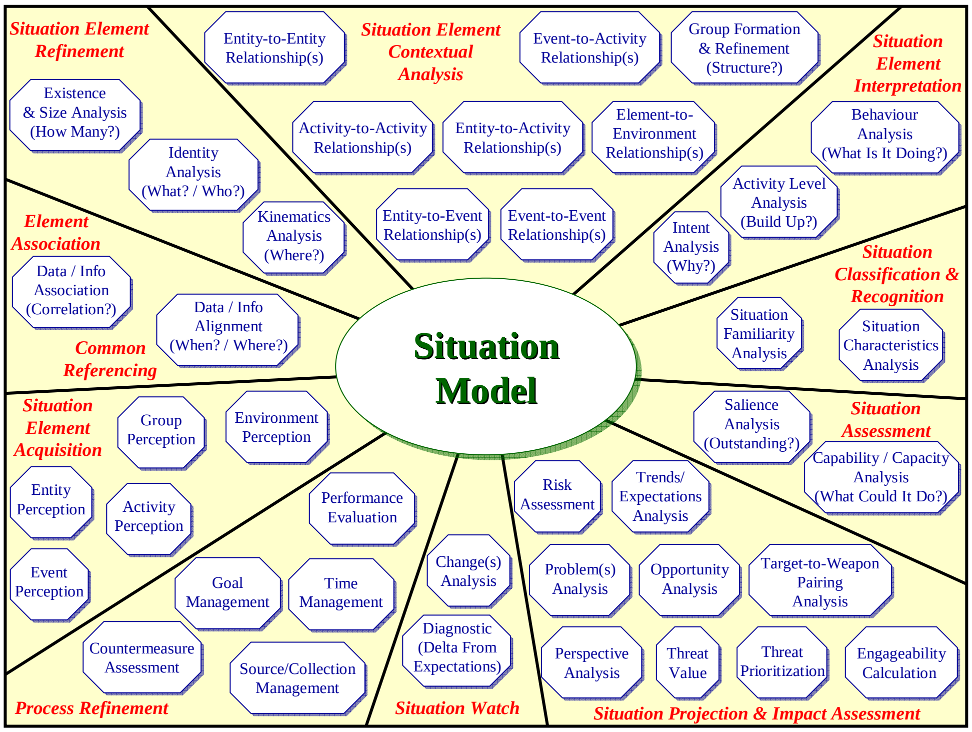

Figure 11: Detailed functional description of the situation analysis process [Roy, 2001]

3.2.1 Situation Model

The main purpose of the SA process is to assemble a representation of aspects of interest in an environment. The SA process thus incorporates and develops an internal situation model of itself and the environment in which the process operate. This situational model, that the SA process endeavours to keep up to date, captures not only the representation of the various elements of the situation, but also a representation of how they relate to create a meaningful synthesis, i.e., a comprehension of the situation. There is one real world, and the situation model is an abstraction of it.

4. Situation Analysis and Knowledge Exploitation Support Systems

This section briefly discusses concepts and issues around the notion of support systems. This is an important aspect as it has been recognized for some time now that situation analysis, data and information fusion, reasoning or inference and decision making as they apply to C4ISR will never be totally automated and embedded in a computer system. The current trend is to rather view the information system technology as supporting the human decision makers. Actually, the challenge is to achieve synergy between the humans and the machines.

4.1 Stress and Pressure - The Need for Technological Support

Operational trends in warfare and public security emergency response activities put the SA process under pressure. For example, in complex military environments known to be non-collaborative, the C2 process is stressed mainly by real-time and uncertainty issues. The technological evolution constantly increases the lethality and the reach of weapons, the scope of the battlefield, and the tempo of the engagement. Moreover, a huge load of uncertain data and information is generated about the environment. Clearly, all these data and information may exceed the human information processing capabilities. Yet, the military community typically maintains that the dominant requirement to counter the threat and ensure the survivability of a military platform is the ability to perform the C2 activities quicker and better than the adversary.

Information technology support is thus typically required to cope with the human limitations in such complex environments. This emphasizes the need for real-time, computer-based support systems (CBSSs) to bridge the gap between the cognitive demands inherent to the accomplishment of the C2 process and the human limitations. Successful exploitation of resources (e.g., sensors, weapons) during the conduct of C2 activities is linked with the modelling and design of systems to support the cognitive demands in a timely manner, using uncertain information.

4.2 The Ideal Support System

A computer-based support system (CBSS) is a computerized system that is intended to interact with and complement a human [Elm *et al.*, 2002]. Such support systems go from formulae embedded within a spreadsheet to sophisticated autonomous reasoning “agents”. Whatever the nature of the CBSS, the objective is to develop CBSS features that intuitively fit the perceptual and cognitive processes of the human user. The ideal CBSS is one that:

- **Provides the information needed by the human decision maker, as opposed to raw data** that must be transformed by the human into the information needed. If the data can be perfectly transformed into information by the CBSS, no human cognitive effort is required for the transformation / no cognitive work is expended on the data, allowing total focus on the domain’s problem solving.
- **Can be controlled effortlessly by the human**. It presents the information to the human as effortlessly as a window allows a view of a physical world outside. In this sense, the CBSS

is “transparent” to the user. If the interaction with the CBSS is completely effortless, no human cognitive effort is required to manage and interact with the CBSS, allowing total focus on the domain’s problem solving.

- **Complements the cognitive power of the human mind**. In this way, the CBSS not only avoids creating a world that is ripe for human decision-making errors but can include features that complement the trends and power of human cognitive processes. Decision-making errors categorized by such terms as fixation, garden path, etc. are avoided by the form of the decision-making world embodied by the CBSS.
- **Supports a wide variety of problem solving strategies**, from nearly instinctive reactions to events to knowledge based reasoning on fundamental principles in a situationally independent manner.

Effective CBSS are the ones that “make the problem transparent to the user”.

4.3 A Generic Support System

Figure 12 shows a generic support system. On the right-hand side are the capabilities supporting the C2 activities per se. On the left-hand side are the necessary ancillary capabilities. All these heterogeneous capabilities are glued together through the core system architecture.

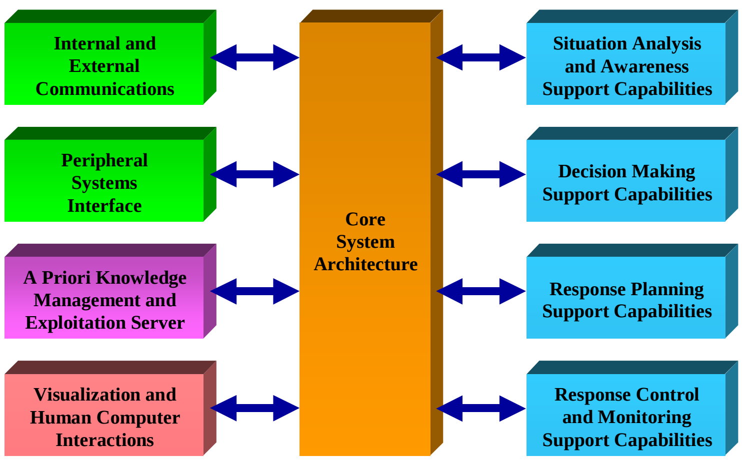

Figure 12: A generic support system

5. Data and Information Fusion

After a short discussion on all information versus the right information, this section provides a definition of data and information fusion, and then present many related concepts.

5.1 Awareness Quality and Decision Making – All Information? The Right Information?

SAW is a key factor determining decision quality. Enhancing SAW improves the probability of selecting the appropriate course of actions in most of the situations. Consequently, SAW is considered essential for commanders to conduct decision-making (DM) activities, and the improvement of the human DM process can be seen as highly related to the enhancement of SAW.

In the same line of thoughts, SAW quality can be related to the amount of information available to an individual. Clearly, circumstances where no information is available should result in poor SAW, leading with high probability to very low decision quality. In such a case, a natural reaction would be to provide mechanisms to increase the amount of information available to the decision makers in order to improve SAW quality. One could even claim that a good approach to reach optimal SAW and DM would be to provide as much information as possible. Along this line of thoughts, many R&D activities are conducted, within DRDC and elsewhere, with the goal of providing "all information, everywhere, at all time".

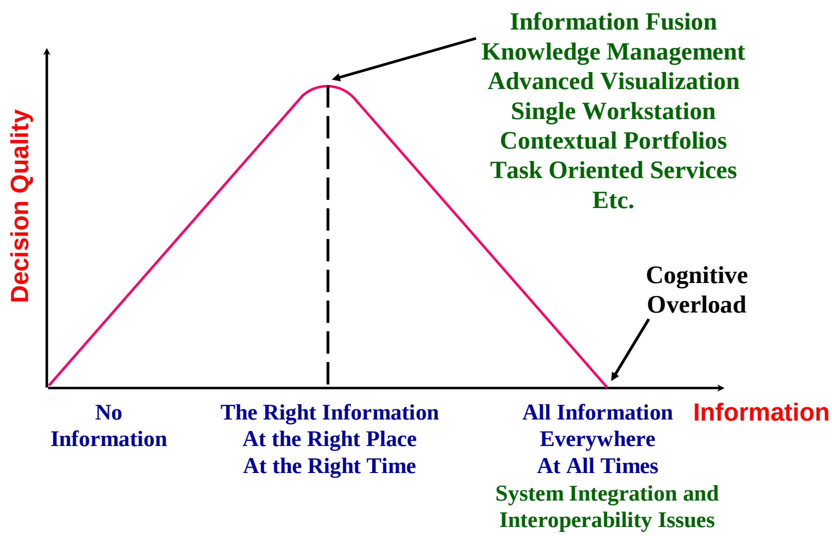

Figure 13: All information? The right information?

Unfortunately, although S&T advancements in information management are necessary, providing "all information, everywhere, at all time" is insufficient and does not necessarily represent the best solution, as more information does not automatically mean better SAW to ensure a better human performance. First, all this information may exceed the human information processing capabilities, leading to cognitive overload. Second, it is not all of the data and information available from the environment that is relevant and useful for reaching an optimal decision. In fact, in some situations, most of the data can be seen as distracters and noise for the decision maker, and may thus reduce his/her level of SAW. The decision maker must detect and use only a specific fraction of this information to enhance his/her SAW and DM processes. Such considerations lead to the concept of “the right information, at the right place, at the right time”, in turn leading to technological enablers such as data and information fusion. This is illustrated in Fig. 13.

Clearly, research and technological advancements toward providing “all information, everywhere, all the time” are necessary, as such progresses ensure that “the right information, at the right place, at the right time” will actually be available to the decision makers. Figure 14 presents a different perspective of a generic computer-based support system that takes into account these issues.

In Fig. 14, the decision-making support capabilities are represented as a set of independent system tools and services supporting situation analysis, decision making, knowledge exploitation, etc. Part of the necessary interactions between these tools/services is enabled by the system integration and interoperability layer. However, the system also requires the appropriate mechanisms, based on technological enablers such as information fusion and knowledge management, to provide “the right information, to the right person, at the right time”.

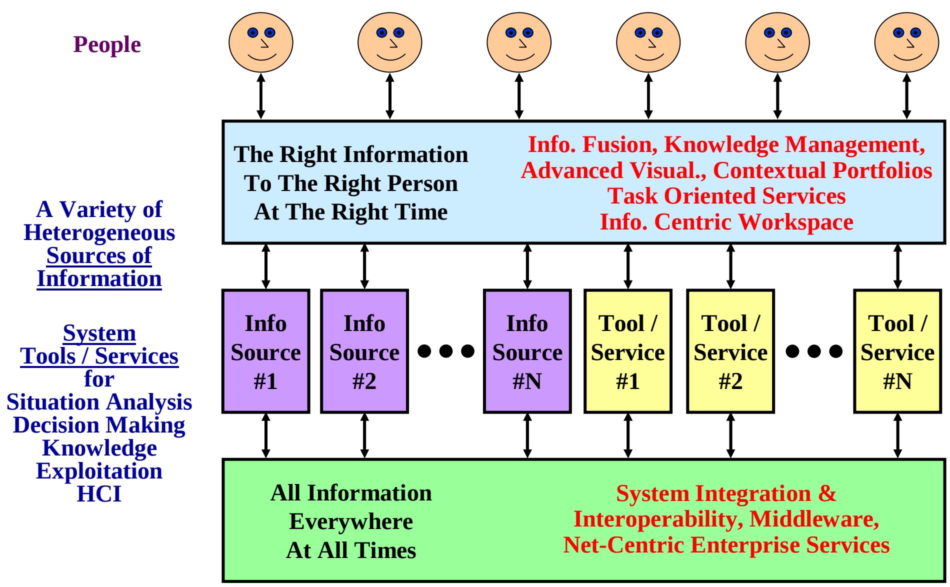

Figure 14: Exploiting information sources and tools/services in a support system

5.2 Data and Information Fusion Definition

Information fusion is a key enabler to meeting the demanding requirements of military and public security command decision support systems, mainly regarding situation awareness. It can be defined as the process of utilizing one or more information sources over time to assemble a representation of aspects of interest in an environment [Lambert, 2001]. This fusion process combines information and refines world state estimates and predictions.

Clearly, data and information fusion has a critical role in achieving situational awareness for future command and control systems. Indeed, it is often asserted that mental information fusion is situation awareness. Fusion allows commanders to cope with the complexity and tempo of operations in the modern dynamic battlespace, and has an important role in asymmetric conflicts. Fusion techniques allow the management of large volumes of information, and the correlation of seemingly unrelated, overlooked, or deceptive information to present a coherent picture of an evolving situation to a decision maker.

5.3 The JDL Data Fusion Model

The data fusion model maintained by the Joint Directors of Laboratories' Data and Information Fusion Group (JDL DIFG) is the most widely-used method for categorizing data fusion-related functions [Steinberg, Bowman, White, 1998]. Figure 15 shows the original model from [White, 1988], while Fig. 16 provides the revised version from Steinberg et al.

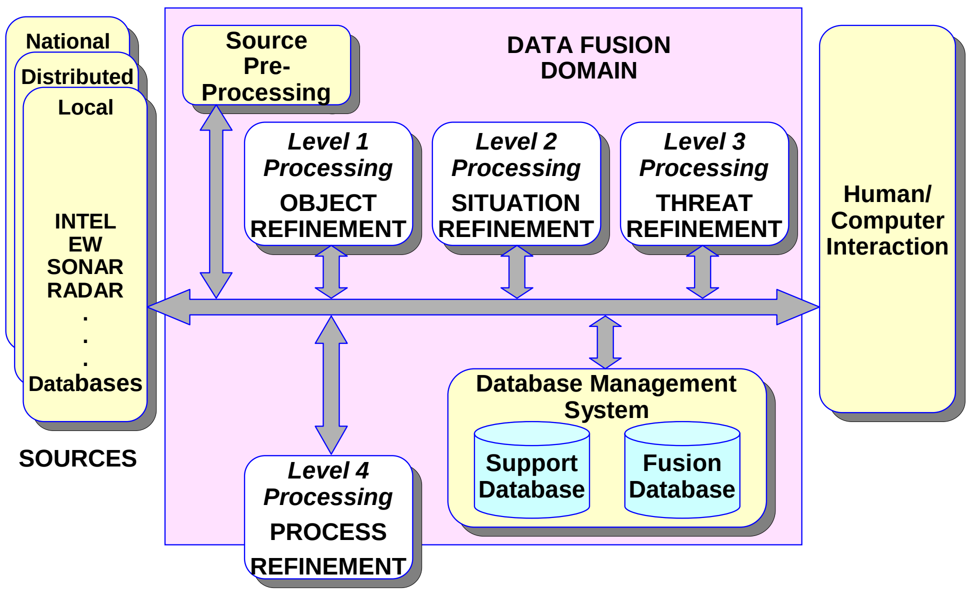

Figure 15: Original data fusion model from the JDL [White, 1988]

The JDL distinction among fusion "levels" provides a valuable way of differentiating between data fusion processes that relate to the refinement of "objects", "situations," "threats" and "processes." The definitions are as follows:

- Level 0 – Sub-Object Data Assessment: Estimation and prediction of signal/object observable states on the basis of pixel/signal level data association and characterization; Level 0 assignment involves hypothesizing the presence of a signal (i.e. of a common source of sensed energy) and estimating its state.
- Level 1 – Object Assessment: Estimation and prediction of entity states on the basis of observation-to-track association, continuous state estimation (e.g., kinematics) and discrete state estimation (e.g., target type and ID); Level 1 assignments involve associating reports (or tracks from prior fusion nodes) into association hypotheses; for which we use the convenient shorthand, 'tracks'. Each such track represents the hypothesis that the given set

of reports is the total set of reports available to the system referencing some individual entity.

- Level 2 – Situation Assessment: Estimation and prediction of relations among entities, to include force structure and cross force relations, communications and perceptual influences, physical context, etc.; Level 2 assignment involves associating tracks (i.e. hypothesized entities) into aggregations. The state of the aggregate is represented as a network of relations among its elements. Any variety of relations is considered – physical, informational, perceptual, organizational – as appropriate to the given system's mission. As the class of relationships estimated and the numbers of interrelated entities broaden, they tend to use the term 'situation' for an aggregate object of estimation.

- Level 3 – Impact Assessment: Estimation and prediction of effects on situations of planned or estimated/predicted actions by the participants, to include interactions between action plans of multiple players (e.g., assessing susceptibilities and vulnerabilities to estimated/predicted threat actions given one's own planned actions); Level 3 assignment is usually implemented as a prediction function, drawing particular kinds of inferences from Level 2 associations. Level 3 fusion estimates the "impact" of an assessed situation; i.e., the outcome of various plans as they interact with one another and with the environment. The impact estimate can include likelihood and cost/utility measures associated with potential outcomes of a player's planned actions.

- Level 4 – Process Refinement: Adaptive data acquisition and processing to support mission objectives. Level 4 processing involves planning and control, not estimation. Level 4 assignment involves assigning tasks to resources.

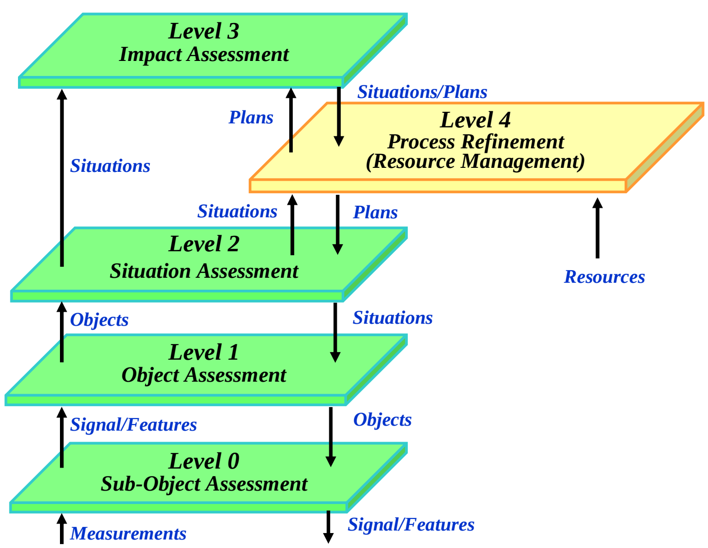

Figure 16: Revised data fusion model [Steinberg, Bowman, White, 1998]

Figure 17 from [Llinas, Antony, 1993] is a good illustration of the overall data fusion process as per the JDL conception (although this figure would need to be updated according to the revised version of the JDL model in [Steinberg, Bowman, White, 1998]). It reflects that in most defence applications, data fusion processing tends to be hierarchical in nature due to the inherent hierarchies built into defence organizations and operations. As a result, the fusion process also progresses through a hierarchical series of inferences at varying levels of abstraction.

Figure 17 also suggests the iterative, continuous nature of these inference processes driven by the temporal character of the usual defence problems. Figure 18 from [Antony, 1995] is also highly representative of the JDL conception of data fusion.

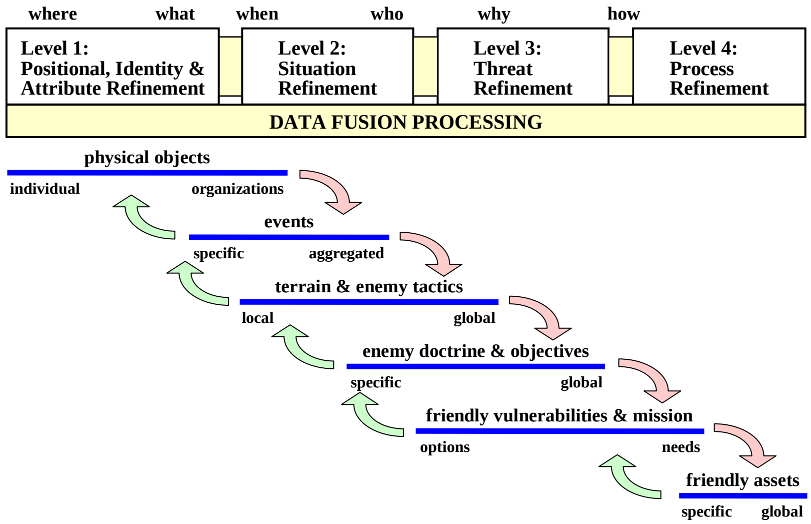

Figure 17: Multi-level/multi-perspective inference [Llinas, Antony, 1993]

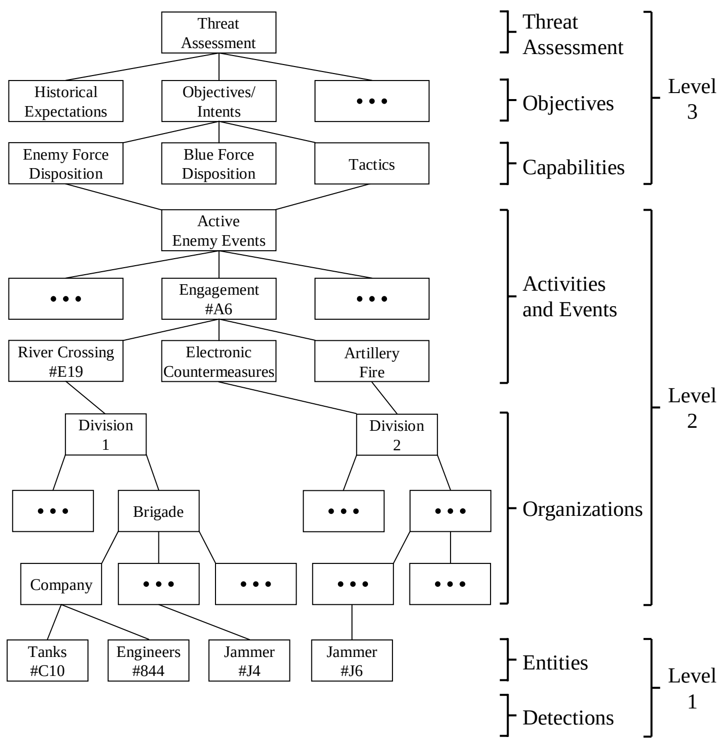

Figure 18: Multiple level-of-abstraction situation view [Antony, 1995]

6. Fusion versus Reasoning/Inference

Not all of the situation elements of interest to a given decision maker are directly observable with the typical data and information sources currently available. This is especially true of highly abstract types of situation elements (e.g., enemy intent), and also of the relationships between the situation elements. Those aspects of interest that cannot be observed must be inferred, i.e., derived as a conclusion from facts or premises, or by reasoning from evidence. This is an essential aspect of situation analysis that will need a lot more S&T attention in the foreseeing future. The user domain ontologies (cf. Section 10 and [Roy, Auger, 2007-A]), derived from ontological engineering [Roy, Auger, 2007-C], and the domain expertise of subject-matter experts, captured and stored in knowledge bases through knowledge engineering activities [Roy, Auger, 2007-C], will be of great help here to support the coherent and consistent analysis of the evidences as received from the sources and accumulated in the system.

Figure 19 shows both notions, i.e., that of a fusion processing node, and that of an inference processing node. This figure also illustrates the distinction between "redundancy" fusion, and "complementary" fusion.

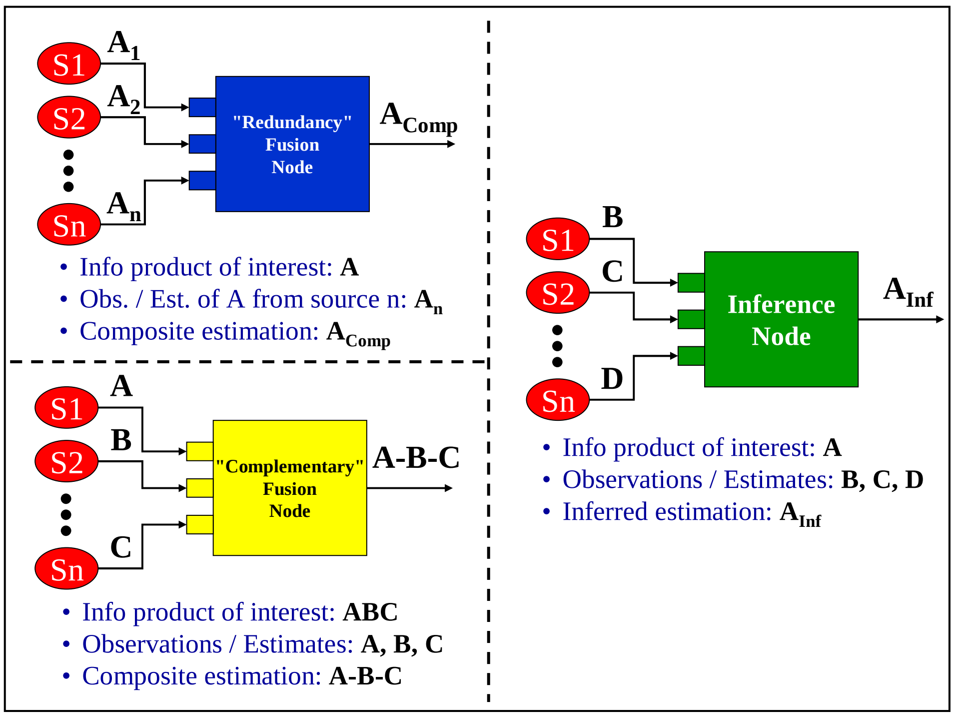

Figure 19: Fusion ("redundancy" and "complementary") and reasoning/inference nodes

6.1 Data/Information Fusion System Nodes

The concept of a "processing node" used in the previous subsection has been introduced in the framework of the JDL fusion model [Steinberg, Bowman, White, 1998], as shown in Fig. 20. According to this fusion node paradigm, the node processes the data/information provided by the sources (or other, prior fusion nodes) at the input, to produce a composite, high quality version of some information products of interest to the users (or to other, subsequent fusion nodes) at the output.

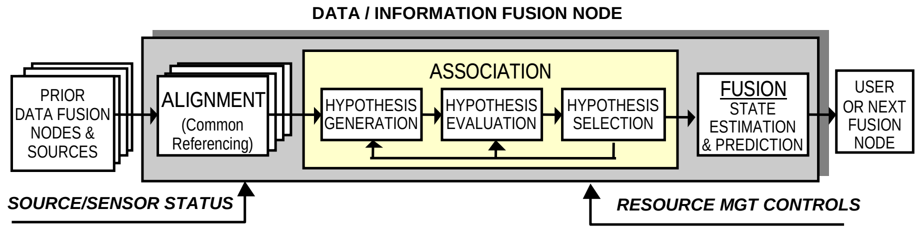

Figure 20: Any data/information fusion node (adapted from [Steinberg, Bowman, White, 1998])

Any data/information fusion node, whatever the fusion level, contains three main sub-processes: alignment, association, and fusion. The means for implementing these functions and the data and control flow among them will vary from node to node and from system to system. Nonetheless, this node paradigm has proven to be a useful model for characterizing, developing, and evaluating fusion systems.

6.2 Knowledge-Based (Expert) System Nodes

The concept of an "inference node" leads to the domain of knowledge-based systems. The term knowledge system is a shorthand for the term knowledge-based system [Stefik, 1995]. A knowledge system is a computer system that represents and uses knowledge to carry out a task. An expert system is an intelligent computer program that uses knowledge and inference procedures to solve problems that are difficult enough to require significant human expertise for their solution [Giarratano, Riley, 1998]. As the applications for the technology have broadened, the more general term knowledge system has become preferred by some people over expert system because it focuses attention on the knowledge that the systems carry, rather than on the question of whether or not such knowledge constitutes expertise [Stefik, 1995]. Figure 21 illustrates the basic concept of a knowledge-based (expert) system.

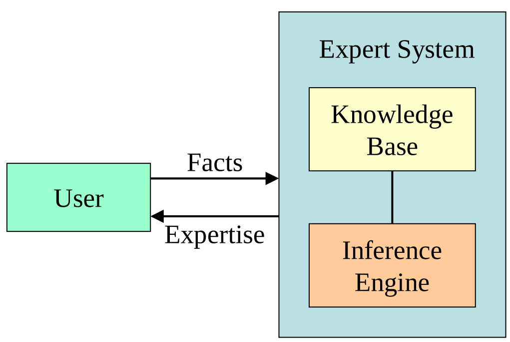

Figure 21: Basic concept of a knowledge-based (expert) system function [Giarratano, Riley, 1998]

The user supplies facts or other information to the expert system and receives expert advice or expertise in response [Giarratano, Riley, 1998]. Internally, the expert system consists of two main components (Fig. 21). The inference subsystem is the part that reasons its way to the solutions of problems, with its search guided by the contents of the knowledge base [Stefik, 1995]. Traditionally, colourfully, and colloquially, this part of a knowledge system has been called the inference engine. It must include provisions for setting goals, representing and recording intermediate results, and managing memory and computational resources. The knowledge base contains the knowledge with which the inference engine draws conclusions [Giarratano, Riley, 1998]. These conclusions are the expert system's responses to the user's queries for expertise.

In a more general way, a knowledge-based system is composed of a knowledge base and an inference mechanism [Russell, Norvig, 1995]. It operates by storing sentences about the world in its knowledge base, using the inference mechanism to infer new sentences, and using them to decide what action to take. One can describe a knowledge-based system at three levels [Russell, Norvig, 1995]:

1. The knowledge level or epistemological level is the most abstract; one can describe the system by saying what it knows.
2. The logical level is the level at which the knowledge is encoded into sentences (assertions).
3. The implementation level is the level that runs on the system architecture. It is the level at which there are physical representations of the sentences of the logical level.

One can also design learning mechanisms that output general knowledge about the environment given a series of percepts. By hooking up a learning mechanism to a knowledge-based system, one can make the system fully autonomous [Russell, Norvig, 1995].

7. A Knowledge-Centric View of Situation Analysis Support Systems

Figure 22 illustrates the main relationships between situation analysis and awareness on one hand, and information fusion and knowledge-based systems on the other hand.

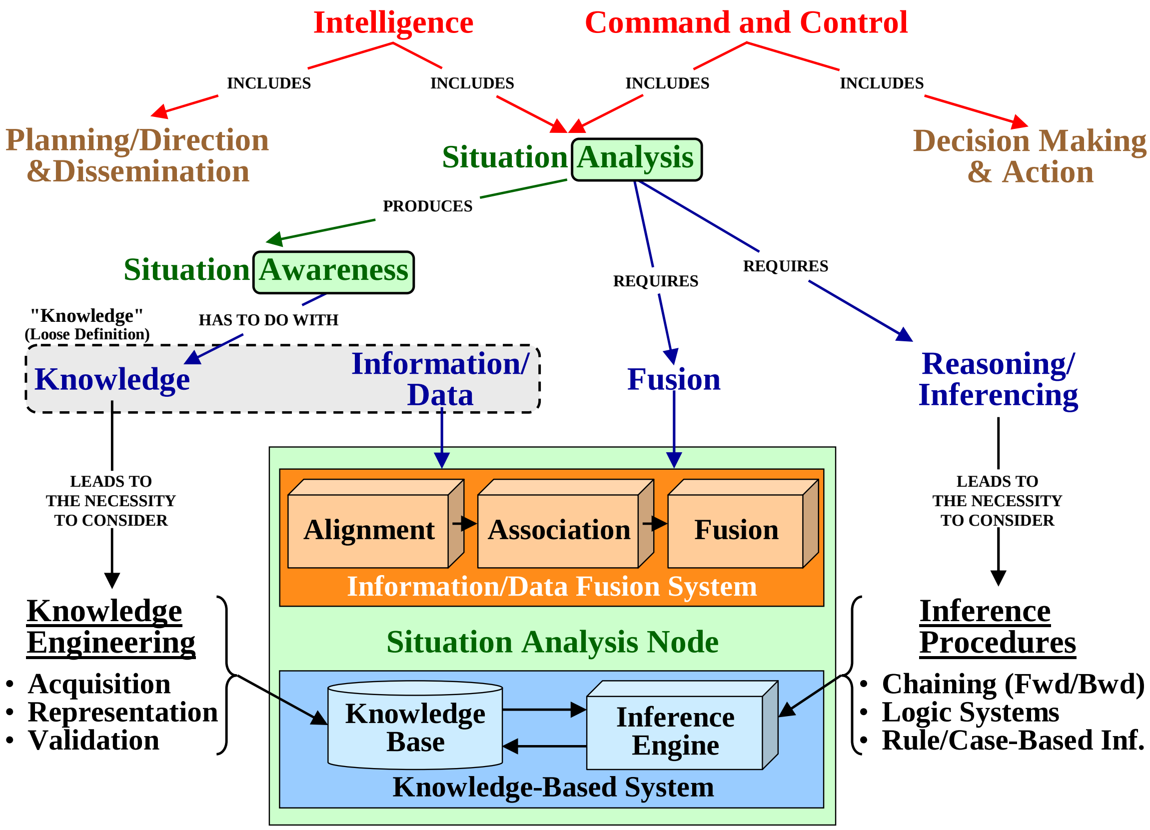

Figure 22: Information fusion and knowledge-based system to support situation analysis

Taking into account the elements illustrated in Fig. 22, the author proposes that developing and adopting a knowledge-centric view of situation analysis should provide a more holistic perspective of this process, leading to the development of better, more adequate SAIF support systems for the operational communities. This is based on a number of notions:

- *Awareness* ultimately has to do with *having knowledge of something*.
- Intelligent agents need knowledge about the world in order to reach good decisions.
- Not all of the situation elements and relationships of interest to a decision maker are directly observable; those aspects of interest that cannot be observed must be inferred, i.e., derived as a conclusion from facts or premises, or by reasoning from evidence.
- There's been a shift of paradigm in Artificial Intelligence (AI) from the pursuit of powerful search and reasoning methods toward a recognition of the role of special case knowledge,

i.e., from a technique-oriented theory of intelligence to a knowledge-oriented theory of intelligence [Stefik, 1995]. The fundamental problem of understanding intelligence is not the identification of a few powerful techniques, but rather the question of how to represent large amounts of knowledge in a fashion that permits their effective use and interaction [Stefik, 1995].

Along this line of thought, the remainder of this report discusses aspects of knowledge, and how it can be formally represented and stored in knowledge bases to be exploited by computer programs. Facets of reasoning are discussed, along with inference methods that can be used in computer applications.

Founded on the material presented and discussed in this report, [Roy, 2007] presents a holistic approach and framework, combining elements of information fusion and knowledge-based systems, for the building of situation analysis support systems. In particular, [Roy, 2007] discusses how knowledge can be acquired from military experts, formally represented, and validated. Knowledge engineering is reviewed, along with cognitive, ontological, and software engineering.

8. Knowledge

Putting knowledge into computers raises many foundational questions [Stefik, 1995]:

- What is knowledge?
- Where does it come from?
- How is it created?
- How is it held by computers?

The study of knowledge is epistemology [Giarratano, Riley, 1998]. It is concerned with the nature, structure, and origins of knowledge. Figure 23 illustrates some of the categories of epistemology.

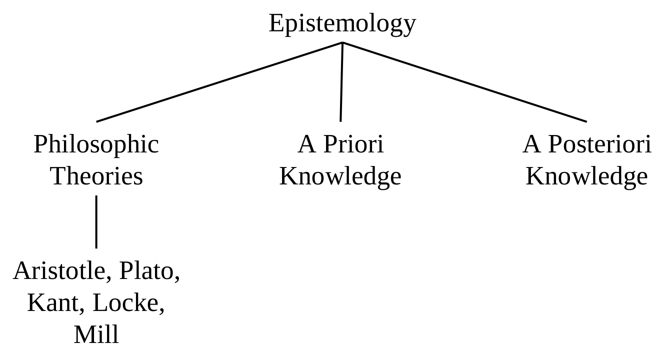

Figure 23: Some categories of epistemology [Giarratano, Riley, 1998]

Besides the philosophical kinds of knowledge expressed by Aristotle, Plato, Descartes, Hume, Kant, and others, there are two special types, called a priori and a posteriori [Giarratano, Riley, 1998]. This chapter further discusses these types of knowledge, along with many other knowledge-related issues.

8.1 Knowledge Definitions

Knowledge is one of those words that everyone knows the meaning of, yet find hard to define [Giarratano, Riley, 1998]. Since the beginning of knowledge engineering, the term *knowledge* has been controversial when it is used as a description of something that computers can represent and use [Stefik, 1995]. Actually, much of the struggle in making sense of knowledge engineering is in dealing with this word. In ordinary usage, the term *knowledge* is used imprecisely and sometimes synonymously with other words such as data, information, and truth. Within technical literature, however, there is potential for confusion and controversy.

Hence, knowledge has many meanings [Giarratano, Riley, 1998]. Knowledge is defined in [Merriam-Webster, 1981] as "*the fact or condition of knowing something with a considerable degree of familiarity through experience, association or contact.*" It has also been defined in [Nonaka, Takeuchi, 1995] as "*a dynamic human process of justifying human belief toward the* *truth.*" Table 1 lists seven meanings of the term "*knowledge*", from a dictionary definition, that are relevant to knowledge systems [Stefik, 1995].

Table 1: Seven meanings of the term "knowledge" [Stefik, 1995]

(1) A clear and certain perception of something; the act, fact, or state of knowing; understanding.	The term perception connotes the certainty provided by reliable sensory perception as portrayed by the expression "seeing is believing". Perception provides evidence about an external reality. All models of knowledge formulation rely on external evidence somehow. Knowledge system development includes feedback loops that ultimately involve the sensory abilities of the system developers and others. However, a strict reliance on perception is too confining and overstates the reliability of the senses. The senses can be fooled. Such strict reliance also ignores the utility of communication for conveying knowledge of distant events. From a perspective of knowledge engineering, perception is a basis for knowledge. Perception plus prior knowledge plus rationality gives a basis for action. but perception should not be confused with knowledge itself.
(2) Learning; all that has been perceived or grasped by the mind.	This can refer either to academic learning or to more mundane forms of everyday learning. The academic meaning is too restricted for knowledge engineering because academic concerns at any given time are only a subset of human concerns. The more mundane interpretation of learning encompasses methods for acquiring information through processes of abstraction, generalization, and model building. In knowledge engineering, machine-learning techniques formulate experience as knowledge. But computers can represent and use knowledge obtained from other agents, without having direct experiences themselves and without generalizing from cases themselves.
(3) Practical experience; skill; as a knowledge of seamanship.	Experience is what knowledge is about and is essential for the creation of knowledge. But one must reflect on the experience to gain knowledge. The seamanship example conveys the idea that to be considered knowledgeable, a person must have a breadth of practical experience. The implication is that a person who has been at sea often enough will probably have encountered enough situations to acquire whatever he needs to know.
(4) Acquaintance or familiarity such as with a fact or place.	Colloquially, we contrast someone who has "book knowledge" with others who have practical experience. A medical intern with book learning may be a riskier candidate for treating a patient than a seasoned doctor. The former's experience is less complete than that of the latter. From a perspective of knowledge engineering, acquaintance and familiarity refer to degrees of knowledge, but should not be

	confused with the nature of knowledge.
(5) Cognizance; recognition.	This refers to a shallow degree of knowledge. If one recognizes a face but cannot remember much about the person, one "knows" the person but not very well. Recognition is often thought to be easier than generation, as in the case of people who can roughly understand a foreign language without being able to speak it. This meaning of the term knowledge is similar in status to meaning #(4).
(6) Information; the body of facts accumulated by mankind.	This suggests that knowledge can be accumulated. This meaning suggests that knowledge is somehow encoded.
(7) Acquaintance with facts; range of awareness, or understanding.	One attributes knowledge to those who demonstrates broad competences. This meaning has the same force and limitations as meaning #(3). It suggests that part of the work of knowing something is being able to apply that knowledge to a range of situations. One must reason in new situations using what one has acquired in specific ones. This implies an ability to infer and to generalize. As in meaning #(3), having knowledge implies broad competence.

Looking back over these dictionary meanings of knowledge, i.e., perception, recognition, learning, experience, competence, it is noteworthy that they are all about relations and processes involving agents in their environment [Stefik, 1995]. Knowledge is not characterized by such properties as weight or extent; it is not a substance or a quantifiable property; it is not simply an encoding. What seems to matter and what one is inclined to describe when one characterizes knowledge are the expectations it creates between environments, agents, and their rational actions. This stance on the meaning of the term *knowledge* is consistent with usage of the term in technical discussions about knowledge systems [Stefik, 1995].

8.1.1 Defining Knowledge in Terms of Situations, Action, and Agents

Knowledge, as the word is used for knowledge-based systems, refers to the codified experience of agents [Stefik, 1995]:

- <u>Codified</u> emphasizes that knowledge is written.
- <u>Experience</u> emphasizes that knowledge is created and used in experiential situations.
- <u>Agents</u> undergo experiences.

The experience is the source of the information for solving problems. By codified, it is meant that the knowledge has been formulated, recorded, and made ready for use. This statement connects the practical intuitions of those who build knowledge systems, the theoretical foundations of knowledge as embedded representations to guide action, and the issues and problems that are driving the development of the field [Stefik, 1995]. The formulation as codified experience acknowledges that such experience is generally hard-won and valued. Thinking about experience and isolating what is new in it is hard work. Experience must be articulated to become explicit knowledge, and that requires more than just a listing of facts. Codified experience must be organized and generalized to guide future action.

One can formalize the process of knowledge creation in terms of the scientific method [Stefik, 1995]. Knowledge is that which is justified by our experience, or more formally, it is what we have learned from our experiments.

It is easy but misleading to overlook the roles of agents with respect to knowledge. When we refer to an "experience", some agent must interact with the world to have the experience [Stefik, 1995]. Usually this agent is assumed to be a person. When we refer to the codification of experience in symbols, some agent must conceive the symbols. One can say that books contain knowledge, but when one does so, one tacitly assumes that there are people who can make sense of the writings in the books. Agents are also involved when one considers written representations of knowledge. When meaning is attributed to systems and symbols, there is necessarily an agent [Stefik, 1995]. Symbols do not have meanings inherently; they are assigned meaning by an observer/agent and the assignment of meaning is in the mind of the observer.

If one says that "Computer X knows that meningitis causes fever", one implies that from the perspective of some observers, the computer system [Stefik, 1995]:

- has representations of meningitis and fever;
- has representations of the relations between them;
- can form judgments about this situation based on generalizations of other situations; and,
- can render and communicate its judgments involving the situations, meningitis as an infectious agent, and fever.

Thus, knowledge cannot be isolated from its creation or use. Experience involves agents in particular situations working on particular tasks with their own background assumptions [Stefik, 1995].

8.2 Knowledge, Information, and Data (KID)

Other words such as data, facts and information are often used interchangeably with knowledge [Giarratano, Riley, 1998]. However, the majority of academics and knowledge management authorities make a distinction between these related, but discrete terms [Girard, 2004]. Table 2 presents this distinction.

Table 2: Distinguishing knowledge, information, and data (KID) [Waltz, 2003]

Data	Individual observations, measurements, and primitive messages from the lowest level of abstraction. Human communication, text messages, electronic queries, or scientific instruments that sense phenomena are the major sources of data. The term *evidence* (data that is determined to be relevant) is frequently used to refer to elements of data.
Information	Organized sets of data. The organization process may include sorting, classifying, or indexing and linking data to place data elements in relational

	context for subsequent searching and analysis.
Knowledge	Information once analyzed, understood, and explained is knowledge, or foreknowledge (predictions or forecasts). Understanding information provides 1) a degree of comprehension of both the static and dynamic relationships of the objects of data, 2) the ability to model structures, and 3) past (and future) behaviour of those objects. Knowledge includes both static content and dynamic processes.

8.2.1 Knowledge Versus Information

Some say that "knowledge is just information" [Stefik, 1995]. Saying that knowledge is just information is like saying that "computer programs are just information" or that "a digital recording of a piano concerto is just information". From a certain perspective, this is correct, but it is not particularly informative and it is dangerous to depend solely on such a view for understanding the nature and importance of knowledge, computer programs, or music.

The term information is most relevant to communicating and storing representations of knowledge [Stefik, 1995]. Information theory as invented by Shannon and others is about the efficient encoding of data for transmission along communication channels. It is concerned with the number of kinds of symbols to be transmitted and with minimal encodings that make it possible for a transmitter and receiver to differentiate among these symbols. Information theory is not concerned with the use or meaning of the symbols. Information theory does not care what symbols mean or how they are used.

8.2.2 Knowledge Versus Data

The distinction between knowledge and data eludes precise technical distinction, but very little of what one calls data is akin to rules of behaviour or knowledge about problem solving [Stefik, 1995]. If knowledge is "codified experience", it should be able to guide action in different situations. Breadth of experience and breadth of applicability matter.

There is an important qualitative difference between, say, a single measurement using a scientific instrument and a thoroughly tested scientific theory [Stefik, 1995]. Both could be used to guide action, but the latter may be suitable for guiding action in many situations. A single measurement is "merely" data in that by itself it does not reflect a substantial body of experience. In this sense, data are less refined than knowledge.

8.2.3 Knowledge/Cognitive Hierarchy

Knowledge, information, and data (KID) can be classified by their degree of abstraction and by their quantity [Turban, Aronson, 1998]. The three terms are hierarchical in nature, with data being the foundation upon which information builds to a pinnacle of knowledge [Girard, 2004]. Hence, these abstractions are often organized in a cognitive hierarchy, where knowledge is presented as one of the top components (often the top component) of a hierarchical pyramid.

Today, several cognitive theories exist that take into account the pyramid of data, information, and knowledge [Girard, 2004]. Some research suggests the hierarchy should extend beyond these three basic building blocks. Systems theorist and professor of organizational change Russell Ackoff sees a hierarchy that extends the pyramid to five levels by adding understanding and wisdom. Verna Allee's Knowledge Archetypes enlarges the original three to seven by adding meaning, philosophy, wisdom, and union. [Giarratano, Riley, 1998] present such a pyramidal view, shown on Fig. 24, of the knowledge/cognitive hierarchy.

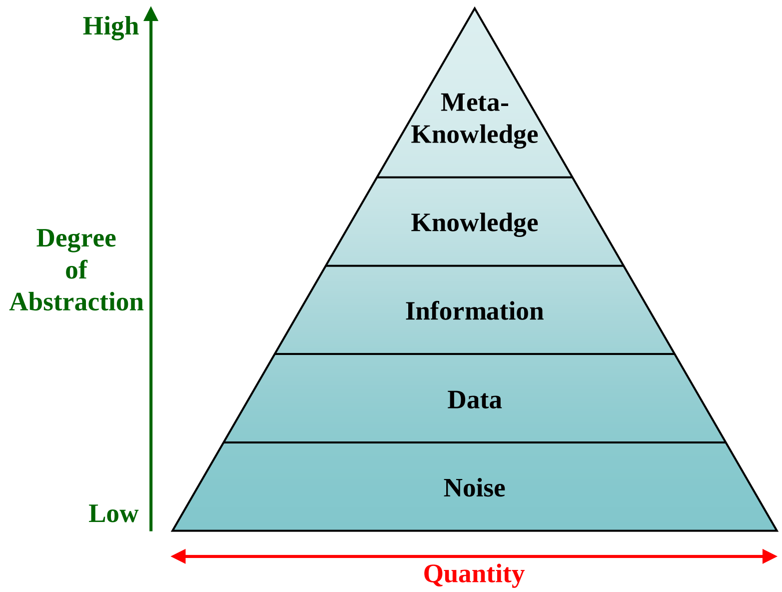

Figure 24: The hierarchy of knowledge [Giarratano, Riley, 1998], [Turban, Aronson, 1998]

At the bottom is noise, consisting of items that are of little interest and that obscure data [Giarratano, Riley, 1998]. Determining what is data and what is noise is like the old saying about gardening, "a weed is anything that grows that isn't what you want". The next higher level is data, which are items of potential interest. Information, or processed data that are of interest are on the third level. The term facts can mean either data or information. Next is knowledge, which represents very specialized information. Knowledge is more abstract and exists in smaller quantity [Turban, Aronson, 1998].

Above knowledge is metaknowledge, i.e., knowledge about knowledge [Giarratano, Riley, 1998]. An expert system may be designed with knowledge about several different domains. Metaknowledge would specify which knowledge base was applicable. Metaknowledge may also be used within one domain to decide which group of rules in the domain is most applicable.

In a philosophical sense, wisdom is the peak of all knowledge [Giarratano, Riley, 1998]. Some consider wisdom to be a uniquely human cognitive capability, the ability to correctly apply knowledge to achieve an objective [Waltz, 2003]. Wisdom is the metaknowledge of determining the best goals of life and how to obtain them [Giarratano, Riley, 1998].

The cognitive hierarchy presented in Table 2 is a very general process definition, neither distinguishing different kinds of knowledge nor making distinctions between two views of knowledge, i.e., as an object or as a process (action) [Waltz, 2003]. One can distinguish between the knowledge-creation processes within the knowledge-creating hierarchy of Fig. 25.

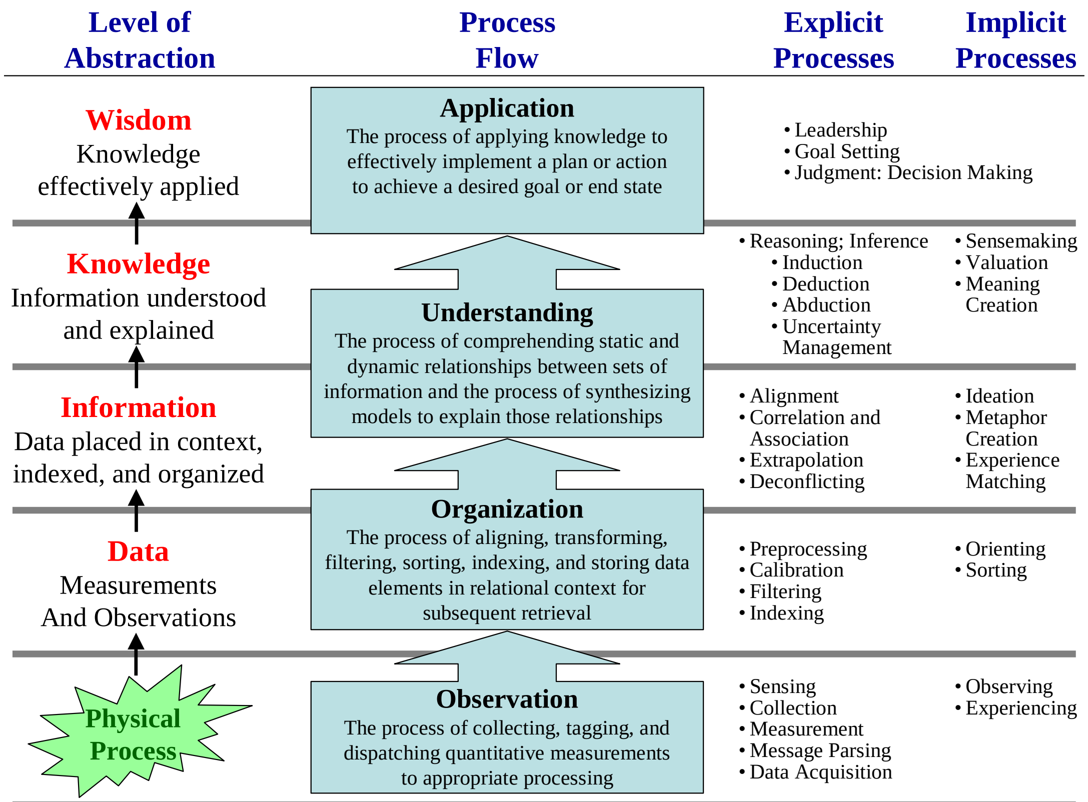

Figure 25: The knowledge-creating hierarchy [Waltz, 2003]

The hierarchy illustrates the distinctions one makes, in common terminology, between explicit (represented and defined) processes and those that are implicit (or tacit; knowledge processes that are unconscious and not readily articulated) [Waltz, 2003].

Occasionally scholars use the collective noun “knowledge” to group together the three blocks of the knowledge-information-data pyramid [Girard, 2004]. For example, in *War and Anti-War*, the futurist authors Alvin and Heidi Toffler use the term knowledge as “defined broadly to include information, data, communication and culture.”. The result of such an unfortunate assemblage is the fallacy that practices such as data processing, information management, and knowledge management are synonymous.

Depending on how they are written, expert systems may draw inferences using data or information [Giarratano, Riley, 1998]. Expert systems may also (1) separate data from noise, (2) transform data into information, or (3) transform information into knowledge.

8.2.4 Military Perspective of the Knowledge/Cognitive Hierarchy

The Canadian Forces (CF) view of the cognitive hierarchy is somewhat different from that of academia [Girard, 2004]. Current doctrine suggests that there are four elements: data, information, knowledge, and understanding.

At the low end are individual sensor observations, or data [Girard, 2004]. This *data* may be processed to develop a common relevant situational awareness, or *information*. Through cognition, one may determine the desired end state, including a commander's intent and concept of operations. This is *knowledge*. At this point, a commander uses judgement to decide what should be done. This is *understanding*, and at this point he or she may take action.

The first three components of this cogitative hierarchy are identical in each of the models, and collectively become the knowledge creation process [Girard, 2004]. Typically, these are the steps completed by the staff to assist the commander. The next step or level of the hierarchy – understanding – is ultimately the domain of the commander. The commander uses his or her judgement to decide on the appropriate action. Unlike the very mechanical knowledge creation steps, this stage tends to be more of an art than a science. The jump from knowledge to understanding builds on the knowledge already created. However, most commanders rely heavily on experience and intuition – in other words more tacit knowledge than explicit knowledge.

Doctrinally, the CF hierarchy culminates at the level of *understanding*; however, there may be merit in exploring Ackoff 's final level, *wisdom* [Girard, 2004]. This almost utopian level presupposes a near perfect understanding of the environment, and tends to be a stage achieved by those few commanders who have extensive experience and finely honed intuition. Those military commanders who mastered the art of warfare truly achieved the "knowledge edge" and were feared by their opponents and heralded by historians. The notion of achieving the knowledge edge in operations is a relatively new concept driven by today's innovative leaders. The knowledge edge was designed primarily with operations in mind; however, it applies equally to other areas of defence.

8.3 Knowledge Categories

Philosophers distinguish several kinds of knowledge, such as "knowledge about what we perceive" [Stefik, 1995]. Forty years ago, Michael Polanyi provided an explanation of knowledge upon which models of knowledge creation have been built [Polanyi, 1967]. He differentiated between explicit, tacit and implicit forms of knowledge [McIntyre, Gauvin, Waruszynski, 2003]. Knowledge can also be further categorized into causal knowledge, shallow or deep knowledge, descriptive knowledge, declarative (factual) or procedural knowledge, a priori or a posteriori knowledge, metaknowledge, etc. These are briefly discussed next.

8.3.1 Explicit Knowledge

Explicit knowledge is that which is stated in detail and leaves nothing merely implied [McIntyre, Gauvin, Waruszynski, 2003]. This is the best known form of knowledge that has been captured and codified in abstract human symbols (e.g., mathematics, logical propositions, and structured and natural language) [Waltz, 2003]. It is tangible, external (to the human), and logical.

This documented knowledge can be stored, repeated, and taught by books because it is impersonal and universal [Waltz, 2003]. It is the basis for logical reasoning and, most important of all, it enables knowledge to be communicated electronically and reasoning processes to be automated.

The development of language, logic, and mathematics has enabled scientific data to be captured, human thought to be recorded, and each to be logically analyzed external to the mind [Waltz, 2003]. Newspapers and novels, HTML content, scientific data, and engineering data all convey explicit knowledge that can be stored, retrieved, and analyzed.

8.3.2 Tacit Knowledge

Tacit knowledge is that which is understood, implied and exists without being stated [McIntyre, Gauvin, Waruszynski, 2003]. This is the intangible, internal, experiential, and intuitive knowledge that is undocumented and maintained in the human mind [Waltz, 2003]. It is a personal knowledge contained in human experience.

For example, an individual knows how to reach with his arm to grasp an object, but cannot describe how he knows how to do it [McIntyre, Gauvin, Waruszynski, 2003]. Other examples are walking or riding a bicycle [Giarratano, Riley, 1998].

This knowledge is unconsciously internalized and cannot be explicitly described (or captured) without effort [Waltz, 2003]. Actually, some say that tacit knowledge is a kind of knowledge that we cannot tell. It is knowledge that cannot be expressed by languages [McIntyre, Gauvin, Waruszynski, 2003]. It is characterized by intangible factors such as perception, belief, values, skill, “gut” feel, intuition, “know-how”, or instinct [Waltz, 2003]. It is thus informal and difficult to capture or share [McIntyre, Gauvin, Waruszynski, 2003].

Some described perception as the “most impoverished form of tacit knowing”, and asserted that there exists higher creative forms of tacit knowing [Waltz, 2003]. This kind of knowledge forms the bridge between perception and the higher forms of (conscious) reasoning that one can tell about more easily. This is the personal knowledge that is learned by experience, honed as a skill, and often applied subconsciously.

8.3.3 Contrasting Explicit and Tacit Knowledge

Explicit and tacit knowledge can be contrasted as two means of knowledge representations, as well as two modes of human thought [Waltz, 2003]. Table 3 presents the bases of explicit and tacit knowledge.

Table 3: The bases of explicit and tacit knowledge [Waltz, 2003]

	Explicit	**Tacit**
Knowledge constructs and modes of human thought	Explicit knowledge represented as an abstraction, context free: Mathematical models and	Tacit knowledge expressed as a narrative, rich in context: Narrative interactive exchanges between storyteller

	logical (context-free) constructs Objective and independent of listener context	and listener Social constructs and experiential context Subjective and dependent upon listener experience
Knowledge description	Physical science; the behaviour and interaction of mass and energy in the material world	Metaphysics; the behaviour and interaction of people, ideas, and minds
Historical basis	Descartes (Discourse on Method) (the physical sciences)	Pascal (Pensées) (metaphysics/the mind)
Knowledge exchange	Objective symbology conveys explicit knowledge	Emotional narration conveys tacit knowledge
Knowledge presenter	Information technology presents objective knowledge in the form of documents, equations, and numerical and graphical visualizations	Humans (knowledge artists or storytellers) describe concepts and perceptions from their own perspective to life to an idea or concept
Protection	Protected by information security (INFOSEC) measures	Protected by operational security (OPSEC) measures

Some have described explicit knowledge as “know-what” and tacit knowledge as “know-how”, distinguishing the ability of tacit knowledge to put explicit knowledge into practice [Waltz, 2003].

The science and mathematics of the Enlightenment Age emerged from the rich development of explicit representations of the physical world [Waltz, 2003]. René Descartes' Discourse on Method is often cited as representative of the basis for this approach of understanding the world. Descartes' reductionist problem-solving method proceeded by [Waltz, 2003]:

- stating assumptions,
- breaking the problem into component parts,
- working on understanding relationships and functions by moving from simple to more complex, and finally,
- integrating the solution into a whole by a logical chain of reasoning.

The Cartesian approach seeks to describe the physical world free of context, objectively, and in pure abstraction.

But tacit knowledge is not of the physical sciences; it is of the mind and the interaction of minds [Waltz, 2003]. For this reason, tacit knowledge is context rich and subjective. In contrast to explicit knowledge of the physical sciences (physics), tacit knowledge, a realm of the mind, is understood in the realm of metaphysics. Pascal, a contemporary of Descartes, is likewise cited as emphasizing the tacit knowledge of the human “heart” [Waltz, 2003]. In his Pensées, Pascal wrote, “The heart has its reasons, which reason does not know”, emphasizing the kind of knowledge that is different than context-free logic.

Explicit knowledge is better understood and represented than knowledge of the tacit kind [Waltz, 2003]. Progress in the cognitive sciences has increased our insight into the capture and representation of this knowledge and the processes underlying its creation, but much has yet to be learned.

Understanding of the processes of exchanging tacit and explicit knowledge will, of course, aid the knowledge management process [Waltz, 2003]. This understanding will enhance the efficient exchange of knowledge between mind and computer, i.e., between internal and external representations.

Some have categorized knowledge in four general classes based on the way in which knowledge is applied [Waltz, 2003]. Table 4 presents these categories.

Table 4: Categories of knowledge [Waltz, 2003]

Category	**Level of Understanding**	**Application Examples**
Explicit content information	Historical record describing the existence, location, and state of physical items and abstract entities	Force inventory Orders of battle Orders, personnel records, intelligence reports
Explicit form information	Static description of the physical shape and composition of objects and characteristics of events	Target models of discriminants for automatic target recognition (ATR) Force model descriptions
Tacit and explicit behavioural knowledge	Experience and dynamic description of the behaviour of an object or system of objects, i.e., behavioural models and simulations	Skills and expertise of subject matter experts (SMEs) Skills and expertise of experienced analysts Weapon simulations Battle management simulation tools
Tacit and explicit actionable knowledge	Insight, experience, and reasoning processes that provide decision-making advice and perform control of operations	Expert judgment of senior intelligence officers Alternative outcomes decision aids Automated sensor management

The categories move from explicit static and dynamic descriptions (models and simulations, respectively) to more tacit representations that are “actionable” [Waltz, 2003]. These categories are helpful to distinguish the movement from explicit representations (data and information) toward mixed tacit and explicit knowledge, which leads to action.

Behavioural knowledge, for example, can be represented in explicit simulations that immerse the analyst in an effort to provide tacit experience with the dynamics of the simulated environment [Waltz, 2003]. The basis of the simulation may be both content and form information about the environment. The result is both tacit and explicit actionable knowledge: insight into alternatives and consequences, risks and payoffs, and areas of uncertainty. All of these form a sound basis for judgment to take action.

8.3.4 Implicit Knowledge

Implicit knowledge is that which could be expressed, but has not been. It is most often thought of as existing within the minds of individuals or in social relationships [McIntyre, Gauvin, Waruszynski, 2003].

8.3.5 Causal, Shallow, and Deep Knowledge

Shallow knowledge is only surface-level information that can be used to deal with very specific situations [Turban, Aronson, 1998]. It may be insufficient in describing complex situations.

Human problem solving is based on deep knowledge of a situation [Turban, Aronson, 1998]. Deep knowledge is the internal and causal structure of a system and considers the interactions among the system's components. Deep knowledge can be applied to different tasks and different situations. It is based on a completely integrated, cohesive body of human consciousness that includes emotions, common sense, intuition, and so on.

Deep knowledge is much more difficult to collect and validate [Turban, Aronson, 1998]. This type of knowledge is difficult to computerize. The system builder must have a perfect understanding of the basic elements and their interactions as produced by nature. To date, such a task has been found to be impossible. Hence, a practical limitation of many expert systems today is lack of causal knowledge, i.e., the expert systems do not really have an understanding of the underlying causes and effects in a system [Giarratano, Riley, 1998]. It is much easier to program expert systems with shallow knowledge based on empirical and heuristic knowledge than with deep knowledge based on the basic structures, functions, and behaviours of objects.

We may never be able to computerize deep knowledge, at its extreme [Turban, Aronson, 1998]. However, it is possible to implement a computerize representation that is deeper than shallow knowledge.

8.3.6 Descriptive Knowledge

Descriptive knowledge relates to a specific object [Turban, Aronson, 1998]. It includes information about the meaning, roles, environment, resources, activities, associations, and outcomes of the object.

8.3.7 Declarative Knowledge

Declarative knowledge is a descriptive representation of knowledge [Turban, Aronson, 1998]. It tells us facts: what things are. It refers to knowing that something is true or false [Giarratano, Riley, 1998]. It is concerned with knowledge expressed in the form of declarative, factual statements. An example is “There is a positive association between smoking and cancer” [Turban, Aronson, 1998]. Domain experts tell us about truths and associations.

This type of knowledge is considered shallow or surface-level information that experts can verbalize [Turban, Aronson, 1998]. It is especially important in the initial stage of knowledge acquisition.

8.3.8 Procedural Knowledge

Procedural knowledge considers the manner in which things work under different sets of circumstances [Turban, Aronson, 1998]. It relates to the procedures used in the problem-solving process (such as information about problem definition, data gathering, the solution process, and evaluation criteria). Hence, it includes step-by-step sequences and how-to types of instructions; it may also include explanations. It is often referred to as knowing how to do something [Giarratano, Riley, 1998]. An example of procedural knowledge is knowing how to boil a pot of water.

Procedural knowledge involves automatic response to stimuli [Turban, Aronson, 1998]. It also may tell us how to use declarative knowledge and how to make inferences.

8.3.9 A Priori Knowledge

A priori knowledge comes before and is independent of knowledge from the senses [Giarratano, Riley, 1998]. As an example, the statements “everything has a cause” and “all triangles in a plane have 180 degrees” are example of a priori knowledge. It is considered to be universally true and cannot be denied without contradiction. Logic statements and mathematical laws are examples of a priori knowledge.

8.3.10 A Posteriori Knowledge

The opposite of a priori knowledge is knowledge derived from the senses, or a posteriori knowledge [Giarratano, Riley, 1998]. The truth or falsity of a posteriori knowledge can be verified using sense experience. However, because sensory experience may not always be reliable, a posteriori knowledge can be denied on the basis of new knowledge without the necessity of contradictions.

8.3.11 Metaknowledge

Metaknowledge is knowledge about knowledge [Turban, Aronson, 1998]. In expert systems, metaknowledge is knowledge about the operation of knowledge-based systems, i.e., about its reasoning capabilities.

8.4 Sources of Knowledge

Knowledge may be collected from many sources [Turban, Aronson, 1998]. It can be identified and collected by using one or several of the human senses, or by machines (sensors, scanners, pattern matchers, intelligent agents, etc.).

Knowledge sources can be divided into two types: documented and undocumented. The latter resides in people's mind. Examples of sources of knowledge include [Turban, Aronson, 1998]:

- human experts,
- textbooks,
- periodicals,
- pictures,
- films,
- multimedia documents,
- maps,
- databases (public and private),
- special research reports,
- flow diagrams,
- stories,
- observed behaviour,
- information available over the world wide web (www),
- etc.

8.5 Knowledge Examples

There is knowledge about natural laws, about social, legal, and regulatory constraints, about effective reasoning in particular contexts, etc. [Stefik, 1995]. Mathematical theorems are arguably formalized knowledge. Other examples of knowledge used in artificial intelligence (AI) are [Turban, Aronson, 1998]:

- facts about a domain,
- concepts,
- general knowledge (e.g., about the world),
- behaviour descriptions,
- vocabulary definitions,
- relationships,
- heuristics,

- decision rules,
- judgement rules,
- procedures for problem solving,
- typical situations,
- hypotheses,
- theories,
- constraints,
- knowledge about knowledge (metaknowledge),
- etc.

Some important types of knowledge are briefly discussed next.

8.5.1 Heuristics

One type of shallow knowledge is heuristic knowledge [Giarratano, Riley, 1998]. *Heuristic* is a Greek term that means “to discover”.

The heuristics express the informal judgemental knowledge in an application area [Turban, Aronson, 1998]. Heuristics are not guaranteed to succeed in the same way that an algorithm is a guaranteed solution to a problem [Giarratano, Riley, 1998]. Instead, heuristics are rules of thumb or empirical knowledge gained from experience that may aid in the solution but are not guaranteed to work.

People often use heuristics, consciously or otherwise, to make decisions [Turban, Aronson, 1998]. By using heuristics, one does not have to rethink completely what to do every time a similar problem is encountered.

In many fields, heuristics play an essential role in some types of problem solving [Giarratano, Riley, 1998]. They are included as a key element of AI in the following definition [Turban, Aronson, 1998]: *AI is the branch of computer science that deals with ways of representing knowledge using symbols rather than numbers and with rule-of-thumb, or heuristics, methods for processing information*. Even if an exact solution is known, it may be impractical to use because of cost or time constraints. Heuristics can provide valuable shortcuts that can reduce both time and cost. For example, many AI methods use some kind of search mechanism [Turban, Aronson, 1998]. Often, heuristics are used to limit the search and focus on the most promising areas.

8.5.2 Global Strategies

Global strategies can be both heuristics and a part of the theory of the problem area [Turban, Aronson, 1998].

8.5.3 Commonsense Knowledge

Commonsense knowledge can be considered to be the knowledge that is generally known [Giarratano, Riley, 1998]. One uses commonsense knowledge when no more situation specific knowledge is available.

8.5.4 Expertise

Expertise is a specialized type of knowledge that is known only to a few [Giarratano, Riley, 1998]. It is not commonly found in public sources such as books and papers. Instead, expertise is the extensive, task-specific and implicit knowledge of the expert that is acquired from training, reading, and experience, and that must be extracted and made explicit so it can be encoded in an expert system. It is usually associated with a vast quantity of knowledge. The following types of knowledge are examples of what expertise includes [Turban, Aronson, 1998]:

- Theories about the problem area.
- Rules and procedures regarding the general problem area.
- Rules (heuristics) of what to do in a given problem situation.
- Global strategies for solving these types of problems.
- Meta-knowledge (knowledge about knowledge).
- Facts about the problem area.

These types of knowledge enable experts to make better and faster decisions than nonexperts in solving complex problems [Turban, Aronson, 1998].

8.5.4.1 Experts

An expert is a person who has (or is recognized by peers as having) expertise in a certain area [Giarratano, Riley, 1998]. This being said, it is sometime difficult to define what an expert is because one actually talks about degrees or levels of expertise [Turban, Aronson, 1998]. The real question is "how much expertise a person should possess before qualifying as an expert?".

In general, the term *expert* connotes both specialization in narrow problem-solving areas or tasks, and substantial competence [Stefik, 1995] [Turban, Aronson, 1998]. Experts are people having substantial training and exceptional skill. They have knowledge or special skills that are not known or available to most people. They have judgement, experience, and methods along with the ability to apply these talents to give advice and solve problems.

An expert can solve problems that most people cannot solve or can solve them more efficiently (but not as cheaply) [Giarratano, Riley, 1998]. Experts can take a problem stated somewhat arbitrarily and convert it to a form that lends itself to a rapid and effective solution [Turban, Aronson, 1998]. Such problem-solving ability is necessary to qualify as an expert, but not sufficient. Typically, human experts possess the following characteristics [Turban, Aronson, 1998], i.e., they:

- recognize and formulate the problem,
- know which facts are important and understands the meaning of the relationships among facts,
- can call up patterns from their experience (excellent recall)
- solve problems quickly and fairly accurately,
- explain the solution and the results, i.e., explain what they do (and sometimes how they do),
- know the extent of their knowledge,
- determine if their expertise is relevant,
- judge the reliability of their own conclusions,
- qualify their advice as the problem reaches their limits of ignorance,
- know when they are stumped,
- degrade gracefully (awareness of limitation),
- communicate with other experts,
- change their points of view to suit a problem,
- restructure knowledge whenever needed,
- break rules whenever necessary (i.e., know the exceptions to the rules),
- learn new things about the domain from experience,
- learn from past successes and mistakes,
- transfer knowledge from one domain to another,
- use tools, such as rules of thumb, mathematical models, and detailed simulations to support their decisions.

Expert activities must be done efficiently (quickly and at low cost) and effectively (with high-quality results) [Turban, Aronson, 1998]. Experts "degrade gracefully", meaning that as they approach the boundaries of their knowledge, they gradually become less proficient at solving problems, but can still develop reasonable solutions. An expert generally can give a measure of how confident he/she is in its solutions when reaching his/her boundaries of knowledge. It takes a long time (usually several years) to become an expert, and novices become experts only incrementally.

8.5.5 Domain Knowledge and Tasks / Problems

A domain is a body of knowledge [Stefik, 1995]. In knowledge representation, a domain is a section of the world about which one wishes to express some knowledge [Russell, Norvig, 1995].

The subject matter of a domain must be recorded or carried somehow [Stefik, 1995]. It may be recorded in written literature or carried by people and conveyed verbally or by apprenticeship training. The term *domain* does not connote anything about the amount of knowledge included.

Often a domain is either a field of academic study or a professional area. Different domains have different degrees of specialization.

A task is a kind of job that is done [Stefik, 1995]. In the context of knowledge systems, a task involves solving a kind of problem. A problem is a particular instance of a task. It consists of a particular set of input data together with the accepted answers. Tasks can be described at different levels of generality.

When one refers to domain knowledge, one means the general terminology and facts of a domain without a focus on a particular task [Stefik, 1995]. When one uses the term task knowledge, one refers to the terminology, computational models, and facts associated with carrying out a kind of task, without necessarily focussing on a particular domain. The term *task domain* combines these ideas. A task domain is the knowledge, assumptions, and requirements associated with doing a particular task in a particular domain. The term is usually used in the context of a particular set of people doing a specialized task.

The first step in solving any problem is defining the problem area or domain to be solved [Giarratano, Riley, 1998]. This consideration is just as true in artificial intelligence (AI) as in conventional programming. An expert's knowledge is specific to one problem domain, as opposed to knowledge about general problem-solving techniques. A problem domain is the special problem area that an expert can solve problems in very well. Expert systems are generally designed to be experts in one problem domain. Expertise in one problem domain does not automatically carry over to another.

The expert's knowledge about solving specific problems is called the knowledge domain of the expert [Giarratano, Riley, 1998]. One should note that the knowledge domain is entirely included within the problem domain. Although general solutions to classic AI problems such as natural language translation, speech understanding, and vision have not yet been found, restricting the problem domain may still produce a useful solution.

8.5.6 Expertise and Domain Knowledge Versus Reasoning

Until the mid-sixties, a major quest of artificial intelligence was to produce intelligent systems that relied little on domain knowledge and more on powerful methods of reasoning [Giarratano, Riley, 1998]. By the early 1970's, it had become apparent that domain knowledge was the key to building machine problem solvers that could function at the level of human experts. Although methods of reasoning are important, studies have shown that experts do not primarily rely on reasoning in problem solving. In fact, reasoning may play only a minor role in an expert's problem solving. Instead, experts heavily rely on a vast knowledge of heuristics and the experience they have built up over the years. If an expert cannot solve a problem based on expertise, then he or she must reason from the first principles and theory. The reasoning ability of an expert is generally no better than that of an average person in dealing with a totally unfamiliar situation. Early attempts at building powerful problem solvers based only on reasoning have shown that they are crippled if they must rely solely on reasoning.

9. Knowledge Management

For a number of reasons, knowledge management (KM) has emerged as an important discipline in the military context:

- the rising importance of having a knowledge advantage over adversaries,
- to attain knowledge superiority,
- to exploit corporate memory,
- to develop the ability to "leverage" defence knowledge.

Knowledge management facilitates the creation and use of knowledge for increased innovation and value [McIntyre, Gauvin, Waruszynski, 2003]. In essence, knowledge organization and human knowledge conversion processes can bring a comprehensive foundation to the common operating picture, interoperability, intelligence, training and acquisitions. As a strategic approach to achieving defence objectives, military KM will play a valuable role in leveraging existing knowledge and converting new knowledge into action through the KM cycle.

After a brief discussion on knowledge as an object versus knowledge as a process, and a review of a well known model of knowledge creation and transfer, this chapter presents some definitions of knowledge management, its enabling disciplines, and the knowledge management cycle.

9.1 Knowledge as an Object Versus Knowledge as a Process

The most common understanding of knowledge is as an object, i.e., the accumulation of things perceived, discovered, or learned [Waltz, 2003]. From this perspective, data (raw measurements or observations), information (data organized, related, and placed in context), and knowledge (information explained and the underlying processes understood) are also objects. The knowledge management field has adopted two basic distinctions in the categories of knowledge as an object: explicit knowledge and tacit knowledge. The terms *resource* and *asset* can be used to describe knowledge, but it is not only an object or a commodity to be managed.

Knowledge can also be viewed as a dynamic, embedded in processes that lead to action [Waltz, 2003]. Knowledge can be viewed as the action, or dynamic process of creation, that proceeds from unstructured content to structured understanding. This perspective considers knowledge as action, i.e., as knowing. Because knowledge explains the basis for information, it relates static information to a dynamic reality. Knowing is uniquely tied to the creation of meaning.

As shown in Table 5, the knowing processes, both explicit and tacit, move from components (data) toward integrated understanding of meaning, i.e., relating the abstractions of knowledge to the real world [Waltz, 2003].

The two paths, though separate columns in the table, are not independent but are interactive [Waltz, 2003]. The explicit knowing process is referred to as reasoning. It is attributed to the Western emphasis on logic, reductionism, and dualism. This knowing process emphasizes the abstraction of truth in the intellect of the individual.

Table 5: Reasoning and sensemaking: knowledge in action [Waltz, 2003]

Knowledge Form and Process	Explicit Reasoning	Tacit Sensemaking	Contribution of the Abstraction
Knowledge	Intellect, hypotheses, explanations, beliefs, education.	Insight, sense, imagination, understandings, perceptions, experience.	Meaning: relative to reality action: basis of action in reality.
Information	Relationships, links, indexed data.	“Images”, metaphors, ideas.	Context: relative to each other
Data	Text, symbolic, numeric.	Experiences, feelings, emotions.	Content: independent abstractions.

By contrast, the tacit knowing process has been called sensemaking, a more holistic form of knowing more closely related to the Eastern emphasis on holistic intuition and oneness [Waltz, 2003]. The term sensemaking has been introduced to describe the tacit knowing process of retrospective rationality, i.e., the method by which individuals and organizations seek to rationally account for things by going back in time to structure events and explanations holistically. One does this to “make sense” of reality, as one perceives it, and create a base of experience, shared meaning, and understanding.

The general knowing process includes four basic phases that can be described in process terms that apply to tacit and explicit knowledge, in human and computer terms [Waltz, 2003]. This is presented in Table 6.

Table 6: Practical transaction processes of knowledge management [Waltz, 2003]

Acquisition	This process acquires knowledge by accumulating data through human observation and experience or technical sensing and measurement. The capture of e-mail discussion threads, digital imaging or signals analysis are but examples of the wide diversity of acquisition methods.
Maintenance	Acquired explicit data is represented in a standard form, organized, and stored for subsequent analysis and application in digital databases. Tacit knowledge is stored by humans as experience, skill, or expertise, though it can be elicitated and converted to explicit form in terms of explanations, stories (rich explanations), or procedures.
Transformation	The conversion of data to knowledge and knowledge from one form to another is the creative stage of knowledge management. This knowledge-creation stage involves more complex processes like internalization, intuition, and conceptualization (for internal tacit knowledge) and correlation and analytic-synthetic reasoning (for explicit knowledge).
Transfer	The distribution of acquired and created knowledge across the enterprise is the fourth phase. Tacit distribution includes the sharing of

	experiences, collaboration, stories, demonstrations, and hands-on training. Explicit knowledge is distributed by mathematical, graphical, and textual representations, from magazines and textbooks to electronic media.

One can also consider the three phases of organizational knowing (focusing on culture) described in Table 7.

Table 7: Three phases of organizational knowing [Waltz, 2003]

Generation	Organizational networks generate knowledge by social processes of sharing, exploring, and creating tacit knowledge (stories, experiences, and concepts) and explicit knowledge (raw data, organized databases, and reports). But these networks must be properly organized for diversity of both experience and perspective and placed under appropriate stress (challenge) to perform. Dedicated cross-functional teams, appropriately supplemented by outside experts and provided a suitable challenge, are the incubators for organizational knowledge generation.
Codification and coordination	Codification explicitly represents generated knowledge and the structure of that knowledge by a mapping process. The map (or ontology) of the organization's knowledge allows individuals within the organization to locate experts (tacit knowledge holders), databases (of explicit knowledge), and tacit-explicit networks. The coordination process models the dynamic flow of knowledge within the organization and allows the creation of narratives (stories) to exchange tacit knowledge across the organization.
Transfer	Knowledge is transferred within the organization as people interact; this occurs as they are mentored, temporarily exchanged, transferred, or placed in cross-functional teams to experience new perspectives, challenges, or problem-solving approaches.

9.2 A Model of Knowledge Creation and Transfer

Knowledge is created and arguably invented [Stefik, 1995]. It does not arise from the passive observation of nature. The idea of “just looking at the world” suggests a naive perspective about what activities one carries out in creating knowledge and what is necessary to generalize, test, and accumulate it.

[Nonaka, Takeuchi, 1995] argue in that effective organizational knowledge creation best occurs through the spiral process where knowledge is converted from tacit to explicit in a continuous and dynamic cycle, as illustrated in Figure 26.

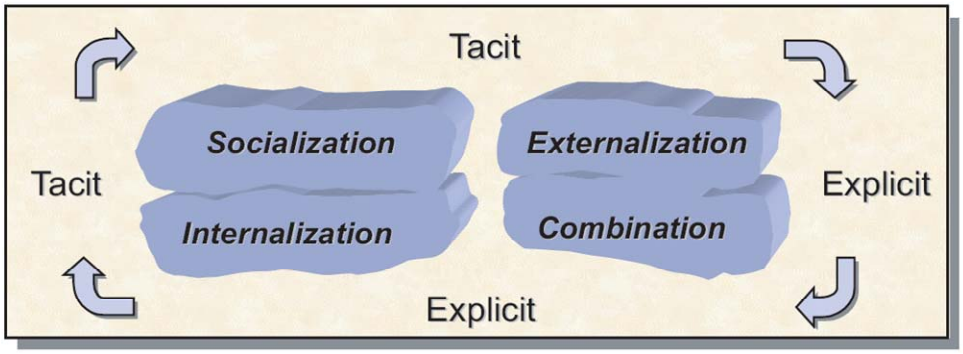

Figure 26: Knowledge creation and transfer cycle [Girard, 2004]

It is when tacit knowledge and explicit knowledge interact that innovation occurs [McIntyre, Gauvin, Waruszynski, 2003]. Knowledge creation is facilitated by deliberately managing the cycle.

Figure 26 is a widely adopted and insightful model of the processes of creating and exchanging knowledge within an organization [Waltz, 2003]. Organizational knowledge creation begins with socialization, where individuals share experience and mental models [McIntyre, Gauvin, Waruszynski, 2003]. It develops into externalization when individuals use metaphors or analogies to articulate hidden tacit knowledge that is otherwise difficult to communicate. It moves into the combination phase for knowledge to be articulated, shared and expounded. Finally, individuals learn by doing and internalizing the new knowledge.

These four modes of conversion operate in an unending spiral sequence to create and transfer knowledge throughout the organization [Waltz, 2003]. The spiral begins again as the experience-based operational knowledge learned in the first cycle provides a larger knowledge base for continuous innovation and growth [McIntyre, Gauvin, Waruszynski, 2003].

The spiral model represents the concept of an ever-learning organization, expanding in knowledge and the application of that knowledge to the dynamic business environment [Waltz, 2003]. It is this model that demonstrates how knowledge is actioned [McIntyre, Gauvin, Waruszynski, 2003].

The four modes of knowledge creation, conversion and transfer, derived from the possible exchanges between two knowledge types, are further discussed next.

9.2.1 Socialization – Tacit to Tacit

Through social interaction, people may gain knowledge that is highly personal and difficult to formalize [Girard, 2004]. Through such interactions, individuals within the organization exchange experiences and mental models, transferring the know-how of skills and expertise [Waltz, 2003]. The primary form of transfer is narrative, i.e., storytelling, in which rich context is conveyed and subjective understanding is compared, “reexperienced”, and internalized. Classroom training, simulation, observation, mentoring, and on-the-job training (practice) build experience; moreover, these activities also build teams that develop shared experience, vision, and values. The socialization process also allows consumers and producers to share tacit knowledge about needs and capabilities, respectively.

One of the best examples is the sharing of experiences through war stories [Girard, 2004]. Properly prepared, these stories are a very powerful way of transferring tacit knowledge from one person to another. When a more experienced soldier, sailor, or air force member recounts a real-life experience to a younger colleague they share more than just the simple facts of the story. Frequently, we witness significant emotions as the veteran shares the difficult and trying conditions lived. The listeners can feel as if they were present at the event.

This process of socialization is an important part of the military heritage, but could be improved [Girard, 2004]. Imagine if all Majors and Master Warrant Officers were trained to tell stories effectively, and this tool were used regularly.

9.2.2 Externalization – Tacit to Explicit

The articulation and explicit codification of tacit knowledge moves it from the internal to external [Waltz, 2003]. This can be done by capturing narration in writing, and then moving to the construction of metaphors, analogies, and ultimately models. Externalization is the creative mode where experience and concept are expressed in explicit concepts. The effort to express is in itself a creative act. This mode is found in the creative phase of writing, invention, scientific discovery, and, for the intelligence analyst, hypothesis creation.

Externalization as a concept of knowledge transfer or creation is foreign to most Western thinkers, with the exception of the military [Girard, 2004]. Within the military construct, one strives to create or transfer tacit knowledge to the explicit form. The lessons learned and after-action review processes are good examples of how military personnel try to codify tacit knowledge.

9.2.3 Combination – Explicit to Explicit

Once explicitly represented, different objects of knowledge can be characterized, indexed, correlated, and combined [Waltz, 2003]. This process can be performed by humans or computers and can take on many forms. Intelligence analysts compare multiple accounts, cable reports, and intelligence reports regarding a common subject to derive a combined analysis. Military surveillance systems combine (or fuse) observations from multiple sensors and HUMINT reports to derive aggregate force estimates. Market analysts search (mine) sales databases for patterns of behaviour that indicates emerging purchasing trends. Business developers combine market analyses, R&D results, and cost analyses to create strategic plans. These examples illustrate the diversity of the combination processes that combine explicit knowledge.

Through the process of codification, one person may document specific knowledge into some form of repository so that many others may access that knowledge [Girard, 2004]. An organization developing and formalizing best practices provides a classic example of the transfer of explicit knowledge. In a defence setting, this is what the military personnel try to achieve using strategy, doctrine and standard operating procedures.

9.2.4 Internalization – Explicit to Tacit

Individuals and organizations internalize knowledge by hands-on experience in applying the results of combination [Waltz, 2003]. Combined knowledge is tested, evaluated, and results in new tacit experience. New skills and expertise are developed and integrated into the tacit knowledge of individuals and teams.

The premise of internalization is knowledge created through an amalgamation of codified explicit knowledge and fuzzy tacit knowledge [Girard, 2004]. An example of this is DND's relatively new value-based ethics program. DND has articulated a series of principles and obligations with the hope that all Defence team members will internalize these important notions and know at an almost instinctive level how to react when confronted with a difficult situation. Over time, the aim is to get most people to base their actions not on a specific obligation, but rather, through the internalization of what they will know is right and wrong.

The internalization mode naturally leads to further socialization, and process leads to further tacit sharing, creativity, and knowledge expansion [Waltz, 2003].

9.3 Some Knowledge Management Definitions

An agreed definition of knowledge management has eluded scholars and practitioners alike since the term first entered our lexicon [Girard, 2004]. Virtually every paper penned on the subject includes a reworked definition, and the debate continues. In the end, it is not the definition that is important but rather the outputs and outcomes of the process. That said, there would appear to be a need for a definition within the DND.

[Waltz, 2003] states that knowledge management refers to the organizational disciplines, processes, and information technologies used to <u>acquire</u>, <u>create</u>, <u>reveal</u>, and <u>deliver</u> knowledge that allows an enterprise to accomplish its mission (achieve its strategic or business objectives). The components of knowledge management are the people, their operations (practices and processes), and the information technology (IT) that move and transform data, information, and knowledge. All three of these components make up the entity we call the enterprise.

[Dieng et al, 1998] state that knowledge management is concerned with the <u>creation</u>, <u>organization</u>, <u>structuring</u>, <u>leveraging</u> and <u>maintenance</u> of knowledge for a more efficient exploitation of the knowledge assets.

Two well-known definitions are representative, although not comprehensively, of how KM is used in corporate management communities [McIntyre, Gauvin, Waruszynski, 2003]:

- The conscious strategy of putting both tacit and explicit knowledge into action by creating context, infrastructure and learning cycles that enable people to find and use the collective knowledge of the enterprise [APQC, 2000].
- The process by which the organization generates wealth from its intellectual or knowledge-based assets [Bukowitz, Williams, 1999].

A recent study within the Department of National Defence suggested that knowledge management in the military varies not in premises or theory from corporate versions, but in terms of context, content and pace [McIntyre, Gauvin, Waruszynski, 2003]. Whereas corporate KM tools can depend on a more sedentary infrastructure, military operational settings require mobile solutions with corresponding issues of security, bandwidth, robustness and reliability. The content varies as well, often more targeted to the particular operation. Finally, most corporate situations do not need the comparable, quick reaction time required in conflict situations. KM in the military context requires [McIntyre, Gauvin, Waruszynski, 2003]:

- knowledge processes that are robust and reliable within operational contexts;
- knowledge content and intellectual assets that are focused, precise, reliable, with suitable recall levels; and
- knowledge creation and conversion processes that match the pace of operations.

A possible definition is proposed in [McIntyre, Gauvin, Waruszynski, 2003] for critique and testing against existing KM initiatives: Military knowledge management is a strategic approach to achieving defence objectives by leveraging the value of collective knowledge through the processes of <u>creating</u>, <u>gathering</u>, <u>organizing</u>, <u>sharing</u> and <u>transferring</u> knowledge into action. It requires processes that are robust and reliable within operational contexts, content and intellectual assets that are focused, precise, reliable, with suitable levels of recall, and knowledge creation and conversion processes that match the pace of operations.

Knowledge management and the knowledge cycle within the context of military operational environments, therefore, require emphasis on these additional requirements of robustness, content and speed. Research and development in the military KM arena must address all components of this definition to be effective [McIntyre, Gauvin, Waruszynski, 2003].

In a 2002 joint letter entitled *Future Directions for Information Management in DND/CF*, the Chief of the Defence Staff (CDS) and the Deputy Minister of National Defence stated that knowledge management was [Girard, 2004]: “An environment that facilitates knowledge <u>discovery</u>, <u>creation</u> and <u>innovation</u>, and which fosters the development of a <u>learning</u> organization.” This description encapsulates the vision of what they expected to see in the future.

According to the Defence Terminology Bank, the draft definition of knowledge management is [Girard, 2004]: “An integrated systematic approach which when applied to an organization enables the optimal use of timely, accurate and relevant information; it also facilitates knowledge <u>discovery</u> and <u>innovation</u>, fosters the development of a <u>learning</u> organization and enhances understanding by integrating all sources of information, as well as individual and collective knowledge and experience.” Like many other knowledge management definitions, this one is somewhat complex.

A concise phrase is desirable – one that immediately communicates the meaning and yet is not constraining. Perhaps a better definition would be simply [Girard, 2004]: “Knowledge management is the <u>creation</u> and <u>sharing</u> of knowledge within Defence”.

At the end of the day, it is up to individual leaders to decide how to create and share knowledge within their organizations [Girard, 2004].

9.4 Enabling Disciplines of Knowledge Management

Knowledge management is a multi-disciplinary field that draws from theories in economics, sociology, philosophy and psychology [McIntyre, Gauvin, Waruszynski, 2003]. Applied disciplines such as information technology, library science and business also contribute to understanding this field. KM combines and applies multiple theories to practical problems within organizations. It has a pragmatic approach that is concerned with real solutions and the ability to analyze and measure its applications accurately.

One authority on the subject, Larry Prusak, writes that knowledge management is rooted in economics and in the need to increase productivity and innovation for economic gain [Prusak, 2001]. KM arose from the economic tenet that productivity is improved through learning, and that continuous improvement occurs through sharing tacit knowledge [McIntyre, Gauvin, Waruszynski, 2003]. Sociology offers insight into social networks and structures as they pertain to knowledge exchange. Psychology provides understanding of human factors and cognitive processes, i.e., how people learn, share, use and create knowledge, and philosophy offers ways of understanding the nature of knowledge itself.

9.5 Knowledge Management Cycle

How knowledge processes in a KM environment are managed to convert knowledge for action and to achieve the desired results of increased value in the organization or specific operations is illustrated in the model of Figure 27 from [Champoux, 1999].

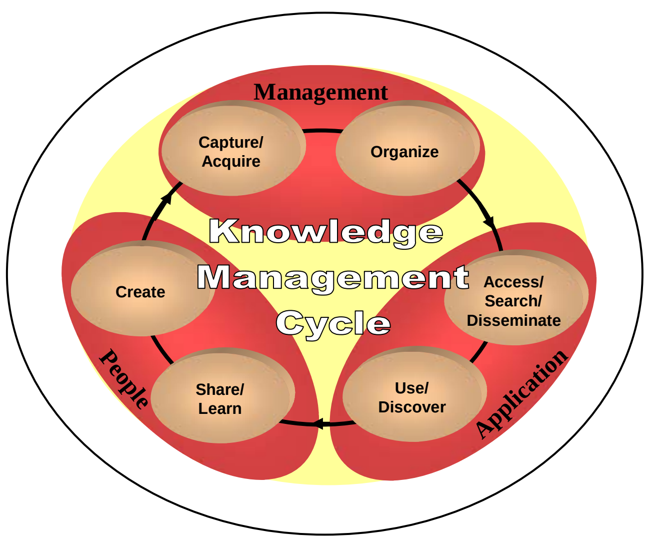

Figure 27: The knowledge management cycle [McIntyre, Gauvin, Waruszynski, 2003]

There are three general perspectives in the cycle: 1) management, 2) application, and 3) people:

- **Management** focuses on capturing, organizing and facilitating knowledge. Many of these activities span the externalization and combination quadrants of the Nonaka model.

- **Application** focuses on effective retrieval of relevant content through advanced searches and mining to conduct knowledge-related work and tasks and on the use of the results for discovery. It relies on the knowledge combination portion of the model.
- **People** focus on learning, sharing and collaboration. This is the education component of the cycle that is within the internalization quadrant, moving into the socialization portion.

Although people, individually and in groups, are part of all perspectives, either as “producers” of background knowledge or as “consumers” of knowledge in the *management* and *application* perspectives respectively, it is within the *people* perspective that their contribution to the collective memory is maximized [McIntyre, Gauvin, Waruszynski, 2003]. Technology may aid them, but in the end, it is their ability to use and innovate with what is available that will create the value realized in KM.

Activities occurring in the cycle are briefly described next.

9.5.1 Capture and Acquire

The drive to capture explicit and tacit forms of knowledge has resulted in the creation of technology tools for creating information repositories and for document and content management [McIntyre, Gauvin, Waruszynski, 2003]. A major challenge in the capture and acquisition of knowledge is to integrate the information collected from a large number of heterogeneous, distributed and disparate “silos”.

9.5.2 Organize

The creation of a KM system requires a structure to organize the content once it is captured [McIntyre, Gauvin, Waruszynski, 2003]. The system must begin with a knowledge model or a meta-model. Models reflect the knowledge components and flows that are inherently embedded in the particular organizational culture and processes. They provide a framework, structure and context to the knowledge base by adding order to the chaos of data, information and knowledge. They also provide the conceptual structure for the design of KM systems and tools. Such modelling is accomplished through the creation of taxonomies, ontologies, semantic networks, glossaries, dictionaries, hierarchies, thesauri, topic maps and metadata.

9.5.3 Access, Search and Disseminate

Effective access, search and dissemination are critically dependent on the organization of knowledge, whether in technological or traditional systems [McIntyre, Gauvin, Waruszynski, 2003]. In technological solutions, search engines are the common applications for these processes. Most are based on full-text indexing using statistical methods (e.g., counting the occurrence and location of words) and on linguistic rules. Alternatives to text indexing are semantic approximation, natural language systems and pattern recognition technologies that make use of semantic functionality to improve the effectiveness and efficiency of retrieving relevant content.

9.5.4 Use and Discover

The potential to extract or share the information in repositories is one of the opportunities of KM technologies [McIntyre, Gauvin, Waruszynski, 2003]. “Knowledge discovery” refers to eliciting knowledge from large data and information sets by identifying new patterns and connections. Within the military setting there is a growing interest in applying technology towards the knowledge management of sense-making, threat analysis and decision making. Applications include visualization, data mining and software agents.

9.5.5 Share and Learn

The results of a recent study conducted by the IBM Institute for Knowledge Management found that even in a company with a well-developed infrastructure of knowledge management technology, people still turn first to other people as they seek solutions to problems and knowledge [McIntyre, Gauvin, Waruszynski, 2003]. Keeping track of who knows what in an organization, particularly a large and geographically dispersed one like the CF, remains a challenge. Social networking allows people to exchange information, and is still one of the most popular means for finding information. Technologies to support knowledge sharing and learning include: portals, web collaboration, smart technologies, e-learning and collaborative intelligence.

9.5.6 Create

It is evident that knowledge creation *per se* is a complex process that involves social and cognitive processes [McIntyre, Gauvin, Waruszynski, 2003]. It is primarily fostered by creating an environment where structure, tools and relationships are made available to the knowledge creators for them to make tacit-tacit, tacit-explicit, explicit-explicit, and explicit-tacit exchanges. When the conversion has occurred, whether it is implicit or codified capture, the cycle returns to the beginning and it is at this point that technological tools can be employed.

9.6 Why Knowledge Management?

The focus thus far has been to define the component and concepts of knowledge management [Girard, 2004]. Using an analogy of the knowledge pyramid, the concentration has been on the data foundation or facts about knowledge management. The next part builds on this important foundation by adding relevance or purpose – in other words, layer two of the pyramid, analogous to information.

Having defined the components of the knowledge pyramid and reviewed knowledge creation and transfer concepts within DND and the Canadian Forces, we may now focus on the question, “Why manage knowledge?” Academic and business leaders alike agree that: “In an economy where the only certainty is uncertainty, the only sure source of lasting competitive advantage is knowledge.” [Girard, 2004]. Experts suggest countless reasons for knowledge management within enterprises, such as globalization, deregulation, technology, downsizing, and information overload.

Though interesting from a professional development point of view, the fact that academia or "Corporate Canada" applies something called knowledge management may only be of passing interest to DND [Girard, 2004]. All too often, the public sector has tended to jump on the bandwagon of the newest hype spawned by business schools. Blind faith in seemingly promising quick-fix solutions can and frequently does lead to unfortunate, though all too predictable, failures. The newest management magic formula has often required significant investment without any benefits, either real or perceived. So why is it that DND and the CF would wish to consider knowledge management?

Surely, this is just another overrated and underdeveloped business process looking for a home, which will unquestionably go the way of the dodo bird [Girard, 2004]. Perhaps, but this seems highly improbable once given the facts. For at the same time Aristotle was considering the categorization of knowledge, Sun Tzu wrote [Girard, 2004]: "If you know your enemy and know yourself, you need not fear the results of a hundred battles." And, at the same time, that knowledge management gurus were selling their wares to business leaders across North America, we heard General Tommy Franks, Commander-in-Chief of U.S. Central Command, saying [Girard, 2004]: "... as has been the case since Sun Tzu said it, precise knowledge of self and precise knowledge of the threat leads to victory."

It is clear that for centuries, enlightened defence leaders, like academics and business leaders, have appreciated the value of knowledge [Girard, 2004]. In fact, there is little need to venture to Iraq in 2003 or Asia in 400 B.C. to hear visionaries speak of the need for knowledge. Embedded in the CF *Strategy 2020* vision statement are the following words of wisdom [Girard, 2004]:

> *The Defence Team will generate, employ and sustain high-quality, combat-capable, inter-operable and rapidly deployable task-tailored forces. We will exploit leading edge doctrine and technologies to accomplish our domestic and international roles in the battlespace of the 21st century and be recognized, both at home and abroad, as an innovative, relevant knowledge-based institution. With transformational leadership and coherent management, we will build upon our proud heritage in pursuit of clear strategic objectives.*

Similarly, the Army Commander's Vision, which serves as a basis for the Army Strategy, includes the following statement [Girard, 2004]:

> *Using progressive doctrine, realistic training and leading-edge technologies, the Army will be a knowledge-based and command-centric institution capable of continuous adaptation and task tailoring across the spectrum of conflict.*

Such passages are not penned without considerable debate [Girard, 2004]. These vision statements are a true reflection of where commanders expect their organizations to be in the future. Clearly, leaders within DND and the CF have acknowledged the importance of moving from the information era to the knowledge age. One need not look very far to see excellent examples of knowledge management in action.

Arguably, DND and the CF constitute one of the most experienced knowledge organizations in Canada and for good reason, given that many knowledge management processes are already commonplace in Defence [Girard, 2004]. For example, DND and the CF have a proven ability to externalize knowledge through the application of lessons learned and after-action review processes. Equally notable are the numerous ways DND and the CF ensure that tacit knowledge is preserved and transferred within the organization, through mechanisms such as battlefield studies and on-the-job training programmes.

The point is that knowledge management is nothing new for Defence. In fact, we trace the origins of knowledge management back to 20 October 1871 [Girard, 2004]. On that day, officers of the first Canadian Permanent Force units met in messes in both Kingston and Quebec. Their meeting spaces are reminiscent of the *ba* concept, a revolutionary knowledge concept being imported from Japan. A *ba* is simply a social space where Japanese executives meet to share knowledge. Such a novel concept is exactly what our forefathers expected of the military mess – a trusted environment where one could share experiences and perhaps tell war stories.

10. Knowledge Representation

One cannot put the world in a computer, so all reasoning mechanisms must operate on representations of facts, rather than on the facts themselves [Russell, Norvig, 1995]. The object of knowledge representation (KR) is to express knowledge in computer-tractable form, such that it can be exploited. KR research studies the problem of finding a language in which to encode knowledge so that the machine can use it [Ginsberg, 1993]. It should support the tasks of acquiring and retrieving knowledge, as well as subsequent reasoning [Turban, Aronson, 1998]. Expert system shells are designed for a certain type of KR such as rules, logic, etc. [Giarratano, Riley, 1998] The way in which an expert system represents knowledge affects the development, efficiency, speed, and maintenance of the system; it thus also has an impact on situation analysis support systems, and ultimately on awareness.

Figure 28 illustrates the main elements involved in KR. The formalization of knowledge in declarative form begins with a conceptualization [Ginsberg, 1993]. Symbols are then used to designate concepts that refer to objects in the world. Symbols are central and familiar elements of spoken and written natural languages, mathematics, logical formalism, and programming languages [Stefik, 1995]. They are fundamental to creating and using computational models. Programs manipulate symbols to solve problems [Turban, Aronson, 1998]. Representation is the assignment of meaning to symbols [Stefik, 1995].

Knowledge systems use a wide range of representations with different expressiveness. The symbols can be combined and arranged into larger structures to express other concepts [Stefik, 1995] [Turban, Aronson, 1998]. The simpler term *expression* is also used to refer to a symbol structure [Stefik, 1995]. A language is a set of expressions and a set of combinatory rules. There are many types of languages; of particular interest are knowledge representation languages.

Roughly, representation languages represent facts in a way that enables their use for many different possible purposes [Stefik, 1995]. A good knowledge representation language should combine the advantages of natural languages and formal languages [Russell, Norvig, 1995]; it should be expressive and concise, be unambiguous and independent of context, and be effective in the sense that there should be an inference procedure that can make new inferences from sentences in the language.

A language is defined by two aspects: syntax and semantics [Russell, Norvig, 1995]; the syntax describes the possible configurations that can constitute sentences, while semantics is an approach for assigning meanings to symbols and expressions. From the syntax and semantics, an inference mechanism can be derived that uses the language. Finally, a representation is formal when its symbols are interpretable by a computer program that uses them to guide its activity in carrying out a task [Stefik, 1995].

One may have the impression that a knowledge engineer must find a single best representation and stick with it [Stefik, 1995]. However, it is not necessary to select and use a single representation in knowledge systems. Actually, no single knowledge representation method is ideally suited by itself for all tasks [Turban, Aronson, 1998]. An important alternative is the use of multiple representations. A variety of knowledge representation paradigms, schemes and techniques have been devised over the years. These includes lists and outlines, decision tables, decision trees, state and problem spaces, production rules, object-attribute-value triples, semantic networks, schemata, frames, scripts, logics, etc.

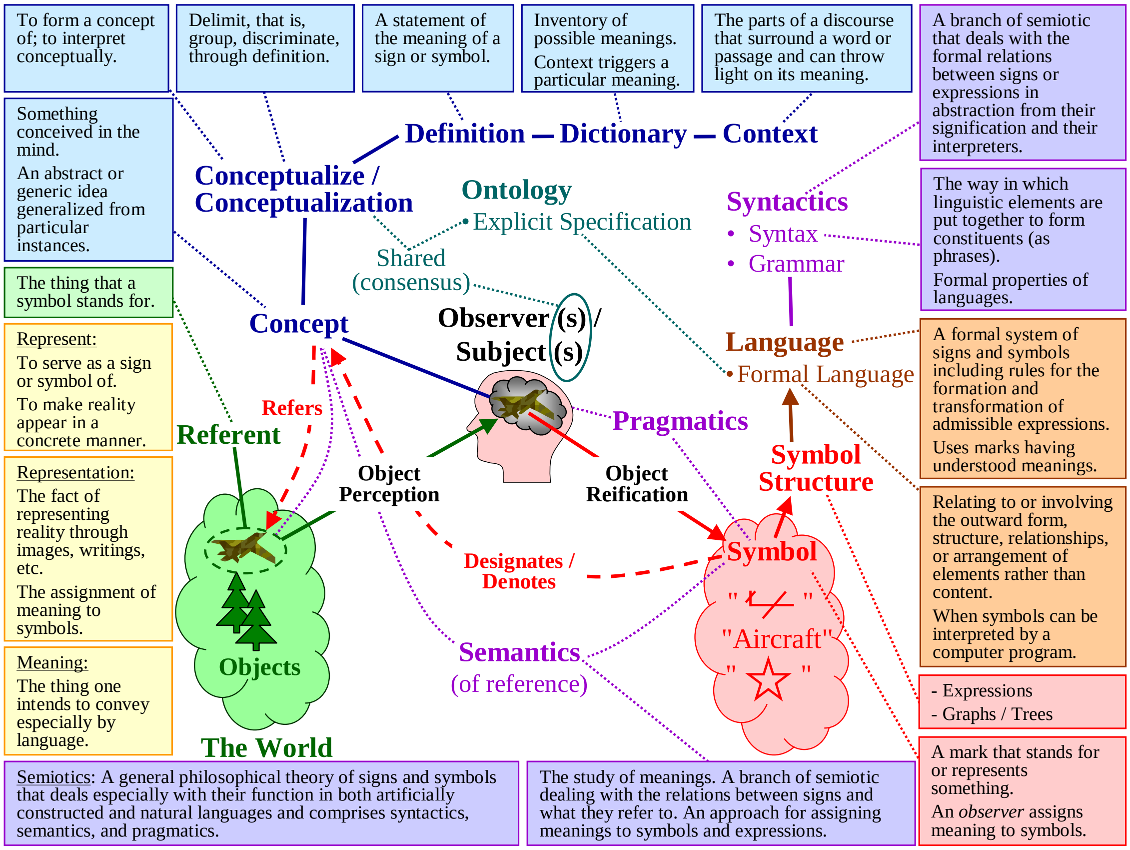

Figure 28: The main elements of knowledge representation

During the last decade, increasing attention has been focused on ontologies and ontological engineering [Gómez-Pérez et al, 2004]. The word *ontology* was taken from Philosophy, where it broadly means a systematic explanation of being. In the last decade, this word has become relevant for the knowledge engineering community, and many definitions of the word have been proposed by a variety of authors. Among these, the definition proposed by [Struder et al, 1998] based on the work of [Gruber, 1993] seems appropriate for the development of knowledge-based Situation Analysis Support Systems (SASSs): “An ontology is a formal, explicit specification of a shared conceptualization.”

This being said, there is actually a whole spectrum of formality that applies to ontologies. They can be highly informal if they are expressed in a natural language, semi-informal if expressed in a restricted and structured form of natural language, semi-formal if expressed in an artificial and formally defined language, and rigorously formal if they provide meticulously defined terms with formal semantics, theorems and proofs of properties such as soundness and completeness [Struder et al, 1998].

There is not only one type of ontology. [Gómez-Pérez et al, 2004] have classified ontologies according to the richness of their internal structure and also to the subject of the conceptualization. Some of the different types of ontologies identified in the literature, based on the subject of their conceptualization, are general (or common), top-level (or upper-level), domain, and task ontologies. Such different types of ontologies can be combined into an ontology library. Regarding generality versus specificity, the reusability-usability trade-off problem applied to the ontology field states that the more general (thus reusable) an ontology is, the less usable it becomes, and vice versa.

Some design criteria and principles that have been proven useful in the development of ontologies are [Gómez-Pérez et al, 2004]: clarity, minimal encoding bias, extendibility, coherence, minimal ontological commitment, representation of disjoint and exhaustive knowledge, minimization of the syntactic distance between sibling concepts, and standardization of names. The ontology community distinguishes lightweight and heavyweight ontologies; lightweight ontologies include concepts, concept taxonomies, relationships between concepts, and properties that describe concepts, while heavyweight ontologies add axioms and constraints.

Finally, different knowledge representation techniques can be applied to model ontologies, although not all of them can represent the same knowledge with the same degree of formality and granularity. One of the key decisions to take in the ontology development process is to select the language (or set of languages) in which the ontology will be implemented.

Discussing all aspects of knowledge representation is clearly out of the scope of this report, as it is a very complex issue, with a multitude of facets. However, a more complete discussion of knowledge representation concepts, paradigms and techniques for use in knowledge-based situation analysis support systems is provided in [Roy, Auger, 2007-A].

11. The Reasoning Processes, Methods and Systems

To reason is [Merriam-Webster, 2003]: 1) to use the faculty of reason so as to arrive at conclusions, or 2) to discover, formulate, or conclude by the use of reason. Similarly, to infer is to derive as a conclusion from facts or premises. The terms *reasoning* and *inference* are generally used to cover any process by which conclusions are reached [Russell, Norvig, 1995]. The term *inference* is generally used for mechanical systems such as expert systems, while *reasoning* is generally used in human thinking [Giarratano, Riley, 1998].

Once the formal knowledge representation in the knowledge base is completed, or is at least at a sufficiently high level of accuracy, it is ready to be used [Turban, Aronson, 1998]. One needs a computer program to access the knowledge for making inferences. This program is usually called the inference engine or the control program. It is used to direct the search through the knowledge base and to control the reasoning process.

A group of multiple inferences that connects a problem with its solution is called a chain [Giarratano, Riley, 1998]. A logical argument is a group of statements in which the last is claimed to be justified on the basis of the previous ones in the chain of reasoning. A theorem is the conclusion of a valid argument. The conclusion of an inference chain is a theorem because it is proved by the chain of inference. Two general method of inference are commonly used as problem-solving strategies: forward chaining and backward chaining.

Forward chaining is reasoning from facts to the conclusion(s) resulting from those facts [Giarratano, Riley, 1998]. It is also called bottom-up reasoning because it reasons from the low-level evidence, facts, to the top-level conclusion(s) based on the facts. It builds up a representation of the situation gradually as new data comes in [Russell, Norvig, 1995]. Its inference processes are not directed toward solving any particular problem; for this reason it is also called a data-driven or data-directed procedure. There are no queries in a forward-chaining approach. Instead, inference rules are applied to the knowledge base, yielding new assertions. This process repeats forever, or until some stopping criterion is met [Russell, Norvig, 1995].

Alternatively, one can start with something one wants to prove, find implication sentences that would allow him/her to conclude it, and then attempt to establish their premises in turn [Russell, Norvig, 1995]. This is called backward chaining, because it uses the inference rules backwards. Backward chaining thus involves “reasoning in reverse”. It is normally used when there is a goal to be proved. It is also called a goal-driven or goal-directed procedure, in which one starts from an expectation (hypothesis), then seek evidence that supports (or contradicts) the expectation [Turban, Aronson, 1998]. Reasoning from the higher-level constructs such as hypotheses down to the lower-level facts which may support the hypotheses is called top-down reasoning [Giarratano, Riley, 1998]. Backward chaining is designed to find all answers to a question posed to the knowledge base [Russell, Norvig, 1995]. Given a query, it searches for a constructive proof that satisfies the query.

Which one is better between forward and backward chaining? The answer depends on the purpose of the reasoning and the shape of the search space [Turban, Aronson, 1998]. For example, if the goal is to discover all that can be deduced from a given set of facts, the system should run forward. In some cases, the two strategies can be mixed (i.e., bidirectional chaining).

Several methods can direct search and reasoning [Turban, Aronson, 1998]. In this regard, it is interesting to examine how people reason, which is what AI attempts to mimic, and there are several ways people reason and solve problems. The most basic taxonomy of inferential reasoning processes distinguishes three basic categories of reasoning [Waltz, 2003]: deduction, induction, and abduction.

Deduction is reasoning about premises to derive conclusions [Waltz, 2003]. One of the most frequently used method of drawing inferences is deductive logic, which has been used since ancient times to determine the validity of arguments [Giarratano, Riley, 1998].

Induction is the method of inference by which a more general or more abstract belief is developed by observing a limited set of observations or instances [Waltz, 2003]. It moves from specific beliefs about instances to general beliefs about larger and future populations of instances. The form of induction most commonly applied is inductive generalization: *All observed As are Bs; therefore, all As are Bs*. Inductive prediction extends belief from a population to a specific future sample: *All observed As are Bs; therefore, the next observed A will be a B.*

Abduction is the informal or pragmatic mode of reasoning to describe how one "reason to the best explanation" in everyday life [Waltz, 2003]. The reasoning process is expressed as a pragmatic syllogism in the following form: *D is a collection of data. Hypotheses* $H_1, H_2, \ldots, H_n$ *all can explain D.* H_k *explains D best. Therefore accept hypothesis* H_k *as the best explanation*.

Some other specific reasoning or inference methods are analogical reasoning, generate-and-test reasoning, model-based reasoning, qualitative reasoning, default reasoning, autoepistemic reasoning, meta-level reasoning, and intuition reasoning. Commonsense knowledge may involve a combination of any of these types [Giarratano, Riley, 1998]. Commonsense reasoning is what people use in ordinary situations, and is very difficult for computers to master.

The following subsection briefly discusses reasoning/inference in computer-based systems. However, as for knowledge representation, discussing all aspects of reasoning processes, methods and systems is clearly out of the scope of this report, as it is a very complex issue, with a multitude of facets. A more complete discussion of these topics is provided in [Roy, Auger, 2007-B].

11.1 Reasoning / Inference Systems

A variety of approaches have been devised and used to achieve reasoning/inference in computer-based systems. Typical examples are logic systems, rule-based systems, frame-based systems, and case-based reasoning systems.

A logic consists of 1) a formal system for describing states of affairs, consisting of the syntax of the language, which describes how to make sentences, and the semantics of the language, which states the systematic constraints on how sentences relate to states of affairs, and 2) the proof theory, i.e., a set of rules for deducing the entailments of a set of sentences. Any logic system has several goals [Giarratano, Riley, 1998]: i) specify the form of arguments, ii) indicate the rules of inference that are valid, and iii) extend itself by discovering new rules of inference and thus extend the range of arguments that can be proved. Numerous logic systems have been developed such as propositional logic, first-order logic, description logic, fuzzy logic, etc. When a logic system is well developed, it can be used to determine the validity of arguments in a way that is analogous to calculations in systems such as arithmetic, geometry, calculus, physics, and engineering [Giarratano, Riley, 1998].

A logical reasoning procedure allows the manipulation of logical expressions to create new expressions [Turban, Aronson, 1998]. It is the process of deriving new sentences from old ones [Russell, Norvig, 1995]. Determining what follows from what the knowledge base contains is the job of the inference mechanism. There are certain patterns of inferences that occur over and over again; such patterns can be captured in what is called inference rules. Modus ponens, an inference schema of a particular propositional form, is the best known rule of inference [Stefik, 1995]. The idea of this rule [Ginsberg, 1993] is that if one knows “If *p*, then *q*”, and if one also knows *p*, then it is legitimate to conclude *q*. Many other rules of inference exist [Stefik, 1995] [Russell, Norvig, 1995] [Ginsberg, 1993].

Rule-based systems use implications (essentially modus ponens) as their primary means for knowledge representation [Russell, Norvig, 1995]. Much of human problem solving or cognition can be expressed by such IF . . . THEN-type production rules [Giarratano, Riley, 1998].

Production rules have two major parts, called the IF-part and the THEN-part [Stefik, 1995]. The IF-part is called by various designations, including the antecedent, conditional part, pattern part, or left-hand-side (LHS) [Giarratano, Riley, 1998]. It consists of conditions to be tested. Following the THEN-part of a rule is a list of actions to be executed when the rule fires. This part of the rule is known as the consequent or right-hand-side (RHS). The consequent of each implication is interpreted as an action recommendation, rather than simply a logical conclusion. Specific actions usually include the addition or removal of facts from the working memory (or insertions and deletions from the knowledge base) as well as input and output (e.g., printing results).

The inference engine in a rule-based system determines which rule antecedents, if any, are satisfied by the facts [Giarratano, Riley, 1998]. When the inference engine notices that a fact satisfies the conditional part of a rule, it puts this rule on the agenda. A rule whose patterns are all satisfied is said to be activated or instantiated. Multiple activated rules may be on the agenda at the same time, in which case the inference engine must select one rule for firing. The inference process may be done in one of two directions, forward or backward, and will continue until no more rules can fire or until a goal is achieved [Turban, Aronson, 1998].

The basic idea of case-based reasoning (CBR) is to adapt solutions that were used to solve old problems and use them to solve new problems [Turban, Aronson, 1998]. CBR finds the cases in memory that solved problems similar to the current problem. It then adapts the previous solution or solutions to fit the current problem, taking into account any differences between the current and previous situations. A case is the primary knowledge-base element for a case-based reasoning application [Turban, Aronson, 1998]. It is an example of a problem that includes the given information, the methods used, and the results obtained [Stefik, 1995]. Finding relevant cases involves characterizing the input problem by assigning appropriate features to it, retrieving the cases with those features from memory, and picking the case or cases that match the input best. CBR has been proposed as a more psychologically plausible model of the reasoning of an expert than a rule-based model [Turban, Aronson, 1998].

12. Artificial Intelligence

Artificial intelligence (AI) is an interdisciplinary field. Many definitions of AI are provided in [Turban, Aronson, 1998], stating that artificial intelligence:

- Is behaviour by a machine that, if performed by a human being, would be called intelligent.
- Is the study of how to make computers do things at which, at the moment, people are better.
- Is basically a theory of how the human mind works.
- Is the branch of computer science dealing primarily with symbolic, nonalgorithmic methods of problem solving.
- Is the branch of computer science that deals with ways of representing knowledge using symbols rather than numbers and with rule-of-thumb, or heuristics, methods for processing information.
- Works with pattern-matching methods that attempt to describe objects, events, or processes in terms of their qualitative features and logical and computational relationships

This being said, most experts agree that AI is concerned with two basic ideas [Turban, Aronson, 1998]:

1. It involves studying the thought processes of humans (to understand what intelligence is, the "Noble laureate" purpose).
2. It deals with representing those processes via machines (such as computers and robots), to make machines smarter (primary goal) and more useful (the entrepreneurial purpose).

The major characteristics of AI are symbolic processing, use of heuristics instead of algorithms, and application of inference techniques [Turban, Aronson, 1998].

Knowledge rather than data or information is the key concept of AI [Turban, Aronson, 1998]. Search methods are also an important aspect of AI. These methods involve solving a problem where the basic technique being used involves examining a large number of possibilities while looking for a solution [Ginsberg, 1993]. Actually, much of the work on AI has to do with the problem of finding ways to attack search problems with the limited computational resources available in practice. AI is thus fundamentally an investigation into knowledge representation and search (as problem solving can commonly be reduced to knowledge representation and search).

12.1 Intelligent Behaviour

As stated above, the primary objective of AI is to build computer systems that perform tasks that can be characterized as intelligent. Several abilities are considered signs of intelligence [Turban, Aronson, 1998]:

- Learn or understand from experience.
- Make sense out of ambiguous or contradictory messages.

- Respond quickly and successfully to a new situation.
- Use reasoning in solving problems and directing conduct effectively
- Deal with perplexing situations.
- Understand and infer in ordinary, rational ways.
- Apply knowledge to manipulate the environment.
- Think and reason.
- Recognize the relative importance of different elements in a situation.

Although AI's ultimate goal is to build machines that mimic human intelligence, the capabilities of current commercial AI products are far from exhibiting any significant success in the abilities just listed [Turban, Aronson, 1998]. Nevertheless, AI programs are continually improving, and they increase productivity and quality by automating several tasks that require some human intelligence.

12.2 Testing for Intelligence

An interesting test designed to determine whether a computer exhibits intelligent behaviour is called the Turing test [Turban, Aronson, 1998]. According to this test, a computer could be considered to be smart only when a human interviewer, conversing with both an unseen human being and an unseen computer, could not determine which is which. Artificial intelligence is the enterprise of constructing an intelligent artifact (a physical-symbol system) that can reliably pass the Turing test [Ginsberg, 1993].

12.3 Artificial Intelligence Versus Natural Intelligence

The potential value of artificial intelligence can be better understood by contrasting it with natural, or human, intelligence [Turban, Aronson, 1998]. This is done here with Tables 8 and 9. Table 8 lists several important advantages of AI.

Table 8: Several important advantages of artificial intelligence [Turban, Aronson, 1998]

AI is more permanent.	Natural intelligence is perishable from a commercial standpoint in that workers can change their place of employment or forget information. However, AI is permanent as long as the computer systems and programs remain unchanged.
AI offers ease of duplication and dissemination.	Transferring a body of knowledge from one person to another usually requires a lengthy process of apprenticeship; even so, expertise can never be duplicated completely. However, when knowledge is embodied in a computer system, it can be copied from that computer and easily moved to another computer, sometimes across the globe.
AI can be less expensive	There are many circumstances in which buying computer

than natural intelligence.	services costs less than having corresponding human power carry out the same tasks.
AI, being a computer technology, is consistent and thorough.	Natural intelligence is erratic because people are erratic; they do not always perform consistently.
AI can be documented.	Decisions made by a computer can be easily documented by tracing the activities of the system. Natural intelligence is difficult to reproduce. For example, a person may reach a conclusion but at some later date may be unable to recreate the reasoning process that led to that conclusion or to even recall the assumptions that were a part of the decision.
AI can execute certain tasks much faster than a human can.	
AI can perform certain tasks better than many or even most people.	

Table 9 lists several important advantages of natural intelligence.

Table 9: Several important advantages of natural intelligence [Turban, Aronson, 1998]

Creativity.	Natural intelligence is creative, whereas AI is rather uninspired. The ability to acquire knowledge is inherent in human beings, but with AI, tailored knowledge must be built into a carefully constructed system.
Use of sensory experience.	Natural intelligence enables people to benefit from and use sensory experience directly, whereas most AI systems must work with symbolic input and representations.
Wide context of experience.	Perhaps most importantly, human reasoning is able to use at all times a wide context of experience and bring that to bear on individual problems. In contrast, AI systems typically gain their power by having a very narrow focus.

Computers can be used to collect information about objects, events, or processes and, of course, computers can process large amounts of information more efficiently than people can [Turban, Aronson, 1998]. However, people instinctively do some things that have been very difficult to program into a computer: They recognize relationships between things, they sense qualities, and they spot patterns that explain how various items relate to each other. One of the ways that human make sense of the world is by recognizing the relationships and patterns that give meaning to the objects and events they encounter. If computers are to become more intelligent, they must be able to make the same kind of associations among the qualities of objects, events, and processes that come so naturally to people.

12.3.1 Despite Limitations of AI

The advantages of natural intelligence over AI show the many limitations of the applied AI technologies [Turban, Aronson, 1998]. However, in many cases AI technologies provide significant improvement in productivity and quality. Although a computer cannot have (as yet) a diversity of experiences, or study and learn as the human mind can, it can use knowledge given to it by human experts. Despite criticisms, AI methods are valuable. They are helping to show how we think and how to better apply our intelligence.

13. Processing / Computing / Programming Paradigms

Artificial Intelligence (AI) is the branch of computer science dealing primarily with symbolic, nonalgorithmic methods of problem solving. This definition focuses on two characteristics [Turban, Aronson, 1998]:

- numeric versus symbolic processing, and,
- algorithmic versus nonalgorithmic processing

13.1 Numerical Versus Symbolic Processing

Computers were originally designed specifically to process numbers (numeric processing) [Turban, Aronson, 1998]. However, people tend to think symbolically; our intelligence seems to be based, in part, on our mental ability to manipulate symbols rather than numbers. When human experts solve problems, particularly the type that are considered appropriate for AI, they do not do it by solving sets of equations or performing other laborious mathematical computations. Instead, they choose symbols to represent the problem concepts and apply various strategies and rules to manipulate these concepts.

The AI approach thus represents knowledge as sets of symbols that stand for problem concepts [Turban, Aronson, 1998]. Symbolic processing is an essential characteristic of AI. However, although symbolic processing is at the core of AI, this does not mean that AI does not involve maths; rather, the emphasis in AI is on the manipulation of symbols.

13.1.1 Physical Symbol Systems

A physical symbol system consists of a set of entities, called symbols, which are physical patterns that can occur as components of another type of entity called an expression (or symbol structure) [Ginsberg, 1993]. Thus, a symbol structure is composed of a number of instances (or tokens) of symbols related in some physical way (such as one token being next to another). At any instant of time the system contains a collection of these symbol structures. Besides these structures, the system also contains a collection of processes that operate on expressions to produce other expressions: processes of creation, modification, reproduction and destruction.

A physical symbol system is a machine, such as a computer, that operates on symbol structures [Stefik, 1995]. It can read, recognize, and write symbols. A symbol system creates new expressions, modifies old expressions, makes new copies of expressions, and destroys expressions. Symbols may be communicated from one part of a symbol system to another in order to specify and control activity. Over time, a physical symbol system produces a changing collection of symbol structures.

13.1.1.1 The Physical Symbol System Hypothesis

The physical symbol system hypothesis states that [Stefik, 1995]:

“A physical symbol system has the *necessary* and *sufficient* means for *general intelligent action*”.

By *necessary*, it is meant that an analysis of any system exhibiting general intelligence would show that the system is a physical symbol system. By *sufficient*, it is meant that any physical symbol system of sufficient complexity could be organized to exhibit general intelligence. By *general intelligent action*, it is meant the same order of intelligent and purposeful activity that we see in people, including activities such as planning, speaking, reading books, or composing music.

This hypothesis puts symbols at the core of intelligent action [Stefik, 1995]. Roughly, it says that intelligent systems are subject to natural laws and that the natural laws of intelligence are about symbol processing.

13.2 Algorithmic Versus Nonalgorithmic Processing

An algorithm is a method of solving a problem in a finite number of steps [Giarratano, Riley, 1998]. More precisely, an algorithm is a clearly defined step-by-step procedure that has well-defined starting and ending points and is guaranteed to reach a solution to a specific problem [Turban, Aronson, 1998]. Most computer architectures readily lend themselves to this step-by-step approach. However, many human reasoning processes tend to be nonalgorithmic; in other words, our mental activities consist of more than just following logical, step-by-step procedures.

13.3 Conventional Versus AI Computing

Conventional computer programs are based on an algorithm [Turban, Aronson, 1998]. It may be a mathematical formula or a sequential procedure that will lead to a solution. The algorithm is converted into a computer program (sequential list of instructions or commands) that tells the computer exactly what operations to perform. The algorithm then uses data such as numbers, letters, or words to solve the problem. Table 10 summarizes some of the ways traditional computers process data.

Table 10: How conventional computers process data [Turban, Aronson, 1998]

Process	Manipulation
Calculate	Perform mathematical operations such as add, subtract, multiply, divide, or find a square root. Solve formulas.
Perform logic	Perform logic operations such as “and”, “or”, or “invert”.
Store	Remember facts and figures in files.
Retrieve	Access data stored in files as required.
Translate	Convert data from one form to another.
Sort	Examine data and put it into some desired order or format.
Edit	Make changes, additions, and deletions to data and change its sequence.

Make structured decisions	Reach simple conclusions based on internal or external conditions.
Monitor	Observe external or internal events and take action if certain conditions are met.
Control	Take charge of or operate external devices.

This process is limited to very structured, quantitative applications.

AI software is based on symbolic representation and manipulation [Turban, Aronson, 1998]. Various processes are used to manipulate the symbols to solve problems, or to generate advice or a recommendation for solving problems.

Once a knowledge base is constructed, some means of using it to solve problems must be developed [Turban, Aronson, 1998]. How does the AI software reason or infer with this knowledge base? The basic techniques are search and pattern matching. Given some initial start-up information, the AI software searches the knowledge base, looking for specific conditions or patterns. It looks for matches that satisfy the criteria set up to solve the problem. The computer literally hunts around until it finds the best answer, based on the knowledge it has. Even though AI problem solving does not take place directly by algorithmic processes, algorithms are used to perform the search process. Table 11 compares AI versus conventional computing.

Table 11: Conventional versus artificial intelligence computing [Turban, Aronson, 1998]

Dimension	Conventional Computing	Artificial Intelligence
Major interest	Data, information	Knowledge
Processing	Primarily algorithmic	Primarily symbolic
Nature of input	Must be complete	Can be incomplete
Nature of output	Must be correct	Can be incomplete
Search	Algorithms	Heuristic (mostly)
Explanation	Usually not provided	Provided
Structure	Control integrated with information (data)	Separation of control from knowledge
Maintenance and update	Usually difficult	Easy because of modularity
Hardware	All types	Mainly workstations and personal computers
Reasoning capability	None	Limited, but improving

13.4 Programming Paradigms and Languages

A language can be defined as a translator of commands written in a specific syntax [Giarratano, Riley, 1998]. Programming paradigms and languages can be classified as procedural and nonprocedural. Figure 29 shows a taxonomy or classification of these paradigms in term of languages.

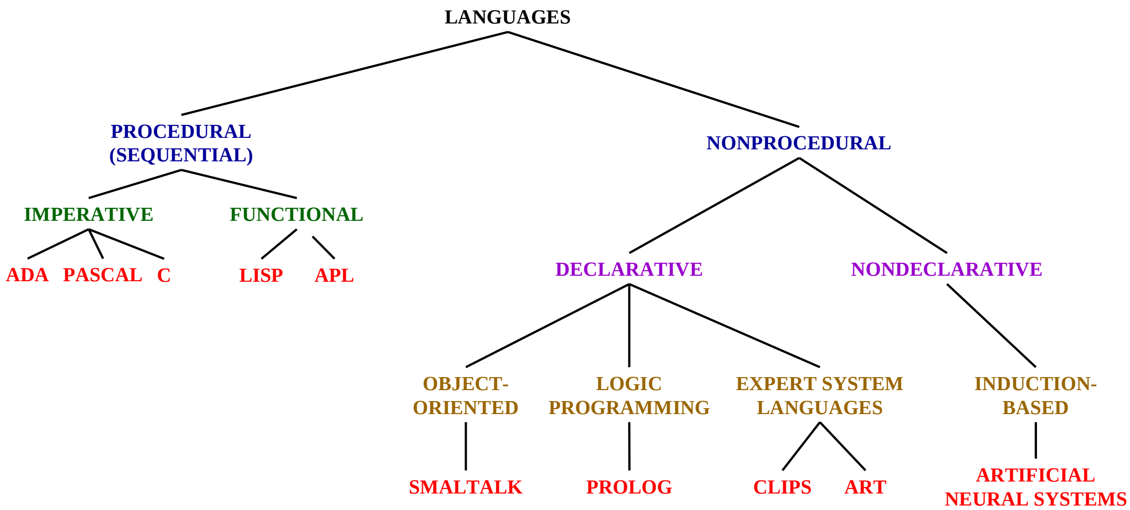

Figure 29: Procedural and nonprocedural languages [Giarratano, Riley, 1998]

This figure should be considered only as a general guide rather than strict definitions. Some of the paradigms and languages have characteristics that may place them in more than one class.

13.4.1 Procedural Paradigm

The implementation of an algorithm in a program is a procedural program [Giarratano, Riley, 1998]. The terms algorithmic programming, procedural programming, and conventional programming are often used interchangeably. A common conception of a procedural program is that it proceeds sequentially, statement by statement, until a branch instruction is encountered. A distinguishing feature of the procedural paradigm is that the programmer must specify exactly how a problem solution must be coded.

13.4.1.1 Imperative Programming

The term imperative and statement-oriented are used synonymously [Giarratano, Riley, 1998]. Statements are imperatives or commands to the computer, telling it what to do. Imperative languages offer great support to variables, assignment operations, and repetitions. Imperative languages are also characterized by their emphasis on rigid control structure and their associated top-down program design.

From the AI standpoint, a serious problem with imperative languages is that they are not very efficient symbol manipulators [Giarratano, Riley, 1998]. Because of their sequential nature, imperative languages are not very efficient for directly implementing expert systems, especially rule-based ones.

13.4.1.2 Functional Programming

The nature of functional programming is different from statement-oriented languages [Giarratano, Riley, 1998]. The fundamental idea of functional programming is to combine simple functions to yield more powerful functions. This is essentially a bottom-up design.

Functional programming is centered on functions [Giarratano, Riley, 1998]. Mathematically, a function is an association or rule that maps member of one set, the domain, into another set, the co-domain. Functional languages are generally implemented as interpreters for ease of construction and immediate user response.

13.4.2 Nonprocedural Paradigm

The nonprocedural paradigm does not depend on the programmer's giving exact details for how a problem is to be solved [Giarratano, Riley, 1998]. The emphasis is on specifying what is to be accomplished and letting the system determine how to accomplish it.

13.4.2.1 Declarative Programming

With the declarative paradigm, the symbols manipulated correspond to objects in the environment [Ginsberg, 1993]. This paradigm separates the goal from the methods used to achieve the goal [Giarratano, Riley, 1998]. The user specifies the goal while the underlying mechanism of the implementation tries to satisfy the goal.

13.4.2.1.1 Object-Oriented Programming

The object-oriented paradigm can be considered partly imperative, and partly declarative [Giarratano, Riley, 1998]. The basic idea of object-oriented design is to design a program by considering the data used in the program as objects and then implementing operations on those objects. Object-oriented design requires no special language features. It can be done in FORTRAN, BASIC, C, and so on. Object-oriented programming thus refers to programming of an object-oriented design even in a language that has no true object support.

13.4.2.1.2 Logic Programming

One of the first AI applications of computers was in proving logic theorems [Giarratano, Riley, 1998]. One of the advantages of logic programming is executable specifications, i.e., specifying the requirements of a problem by Horn clauses produces an executable program, which differs from conventional programming, in which the requirements document does not look at all like the final executable code.

13.4.2.1.3 Expert Systems Languages

Although expert systems are a branch of AI, there are specialized languages for expert systems that are quite different from the commonly used AI languages such as LISP and PROLOG [Giarratano, Riley, 1998]. An expert system language is a higher-order language than languages like LISP or C because it is easier to do certain things, but there is also a smaller range of problems that can be addressed. The primary functional difference between expert system languages and procedural languages is the focus of the representation.

Procedural languages focus on providing flexible and robust techniques to represent data [Giarratano, Riley, 1998]. For example, data structures such as arrays, records, linked lists, stacks, queues, and trees are easily created and manipulated. Modern languages are designed to aid in data abstraction by providing structures for encapsulation. This provides a level of abstraction that is then implemented by methods such as operators and control statements to yield a program. The data and methods to manipulate it are tightly interwoven.

In contrast, expert system languages focus on providing flexible and robust ways to represent knowledge [Giarratano, Riley, 1998]. The expert system paradigm allows two levels of abstraction: data abstraction and knowledge abstraction. Expert system languages specifically separate the data from the methods of manipulating the data. An example of this separation is that of facts (data abstraction) and rules (knowledge abstraction) in a rule-based expert system language.

The difference in focus between procedural and expert system languages leads to a difference in program design methodology [Giarratano, Riley, 1998]. Because of the tight interweaving of data and knowledge in procedural languages, programmers must carefully describe the sequence of execution. In contrast, the explicit separation of data from knowledge in expert system languages requires considerably less rigid control on the execution sequence. Typically, an expert system language will also provide an entirely separate piece of code, the inference engine, which is used to execute the statements of the language and to apply the knowledge to the data. This separation of knowledge and data allows a higher degree of parallelism and modularity. Depending on the implementation, the inference engine may provide forward chaining, backward chaining, or both.

Expert systems can thus be considered declarative programming because the programmer does not specify how a program is to achieve its goal at the level of an algorithm [Giarratano, Riley, 1998]. In a rule-based expert system, any rule may become activated and put on the agenda if its LHS matches the facts. The order in which the rules were entered does not affect which rules are activated. Thus, the program statement order does not specify a rigid control flow.

Table 12 lists some typical differences between expert systems and conventional programs.

Table 12: Typical differences between expert systems and conventional programs [Giarratano, Riley, 1998]

Characteristic	**Conventional Program**	**Expert System**
Control by . . .	Statement order	Inference engine
Control and data	Implicit integration	Explicit separation
Control strength	Strong	Weak
Solution by . . .	Algorithm	Rules and inference

Solution search	Small or none	Large
Problem solving	Algorithm is correct	Rules
Input	Assumed correct	Incomplete, incorrect
Unexpected input	Difficult to deal with	Very responsive
Output	Always correct	Varies with problem
Explanation	None	Usually
Applications	Numeric, file, and text	Symbolic reasoning
Execution	Generally sequential	Opportunistic rules
Program design	Structured design	Little or no structure
Modifiability	Difficult	Reasonable
Expansion	Done in major jumps	Incremental

13.4.2.2 Nondeclarative Programming

Nondeclarative programming are becoming popular [Giarratano, Riley, 1998]. In the induction-based programming paradigm, the program learns by example. The Artificial Neural System (ANS) programming paradigm is based on how the brain processes information. This paradigm is sometimes called connectionism because it models solutions to problem by training simulated neurons connected in a network.

14. Expert Systems

Expert systems are a branch of Artificial Intelligence (AI) that makes extensive use of specialized knowledge to solve problems at the level of a human expert [Giarratano, Riley, 1998]. They are the paradigmatic application of AI techniques to hard problems [Ginsberg, 1993].

The idea is to find a domain in which there are recognized experts, and then to scrutinize those experts to draw out the specialized knowledge that they use when solving problems in their restricted domain [Ginsberg, 1993]. Machines can then (in principle, at least) mimic the reasoning of the experts to derive conclusions similar to theirs using the information with which they have been supplied. Expert systems can thus provide direct means of applying expertise and are used more than any other applied AI technology [Turban, Aronson, 1998]. The area of expert systems is a very successful approximate solution to the classic AI problem of programming intelligence [Giarratano, Riley, 1998].

Drawing from [Giarratano, Riley, 1998] and [Turban, Aronson, 1998] for a definition, an expert system is an intelligent advisory computer program that uses human knowledge captured in a computer and inference procedures to solve specific problems that are difficult enough to ordinarily require significant human expertise for their solution. An expert system is thus a computer program whose performance is guided by specific expert knowledge in solving problems [Stefik, 1995]. The problem-solving focus is crucial in this characterization. The knowledge of central interest in expert systems is that which can guide a search for solutions. The term expert connotes both narrow specialization and substantial competence. Although the term expert has been loosely applied in some cases, it is intended to describe systems that solve problems that are otherwise solved by people having substantial training and exceptional skill.

Well-designed systems imitate the reasoning processes experts use to solve specific problems [Turban, Aronson, 1998]. In the knowledge domain that it knows about, the expert system reasons or make inferences in the same way that a human expert would infer the solution of a problem [Giarratano, Riley, 1998]. That is, an expert system is a computer system that emulates the decision-making ability of a human expert. The term “emulate” means that the expert system is intended to act in all respect like a human expert. An emulation (to imitate, to strive to equal or approach equality with) is much stronger than a simulation (to give or assume the appearance or effect of). The standard of performance for expert systems is thus in human terms, by comparison with people carrying out a particular kind of task [Stefik, 1995]. Ultimately, expert systems could function better than any single human expert in making judgements in a specific, usually narrow, area of expertise (referred to as a domain) [Turban, Aronson, 1998].

This being said, the purpose of an expert system is not to replace the experts, but to make their knowledge and experience more widely available [Turban, Aronson, 1998]. Experts get sick or become unavailable, so knowledge is not always available when needed. Expert systems are the most common technology used to substitute for human expertise, by providing the necessary knowledge. The basic objective with an expert system is to transfer expertise from an expert to a computer system and then to other humans (nonexperts). Expert systems are used to propagate scarce knowledge resources for improved, consistent results. An expert system permits nonexperts to increase their productivity, improve the quality of their decisions, and solve problems when an expert is not available. Because of their ability to enhance productivity and to augment workforces in many specialty areas where human experts are becoming increasingly difficult to find and retain, expert systems are of great interest to organizations.

Expert systems are generally designed differently from conventional programs because the problems usually have no algorithmic solution and rely on inferences to achieve a reasonable solution [Giarratano, Riley, 1998]. Note that a reasonable solution is about the best one can expect if no algorithm is available to help him/her achieve the optimum solution. Depending on the input data and the knowledge base, an expert system may come up with the correct answer, a good answer, a bad answer, or no answer at all. While this may seem shocking at first, the alternative is no answer all the time.

14.1 Expert Systems and Knowledge-Based Systems

The insight that domain knowledge was the key to building real-world problem solvers led to the success of expert systems [Giarratano, Riley, 1998]. The successful expert systems today are thus knowledge-based expert systems rather than general problem solvers.

In this line of thoughts, the terms expert system, knowledge-based system, or knowledge-based expert system are often used synonymously [Giarratano, Riley, 1998]. When expert systems were first developed in the 1970's, they contained expert knowledge exclusively. However, the term expert system is often applied today to any system that uses expert system technology. The knowledge in expert systems may be either expertise or knowledge that is generally available from books, magazines, and knowledgeable persons. It is common practice today to use the term expert systems when referring to both expert systems and knowledge-based systems, even when the knowledge is not at the level of a human expert. This being said, the expression knowledge-based system is a better term for the application of knowledge-based technology, which may be used for the creation of either expert systems or knowledge-based systems.

14.2 Knowledge Representation in Expert Systems

The knowledge of an expert system may be represented in a number of ways [Giarratano, Riley, 1998]. One common method of representing knowledge is in the form of IF . . . THEN rules. One of the major roots of expert systems is the area of human information processing called cognitive science. The study of cognition is very important if one wants to make computers emulate human experts. Cognitive psychologists have used rules as models to explain human information processing. The basic idea is that sensory input provides stimuli to the brain. The stimuli trigger the appropriate rules of long-term memory which produces the appropriate response.

In designing an expert system, an important factor is the amount of knowledge or granularity of the rules [Giarratano, Riley, 1998]. Too little granularity makes it difficult to understand a rule without reference to other rules. Too much granularity makes the expert system difficult to modify, because several chunks of knowledge are intermingled in one rule.

Many significant expert systems have been built by expressing the knowledge of experts in rules [Giarratano, Riley, 1998]. However, while rules are a popular paradigm for representing knowledge, other types of expert systems use different representations. Some types of expert system tools (such as CLIPS) allow objects as well as rules. Rules can pattern match on objects as well as facts. Alternatively, objects can operate independently of the rules.

14.3 Elements of a Rule-Based Expert System

This section discusses the typical main parts of a knowledge-based expert system. Note, however, that an inventory of program parts is not the best way to understand a knowledge system [Stefik, 1995]. The basic concept of a knowledge-based expert system function (or processing node) has already been discussed and illustrated in Fig. 21, in subsection 6.2. Going into more details here, a rule-based expert system typically consists of the components listed in Table 13. These components of a typical rule-based expert system are shown in Fig. 30.

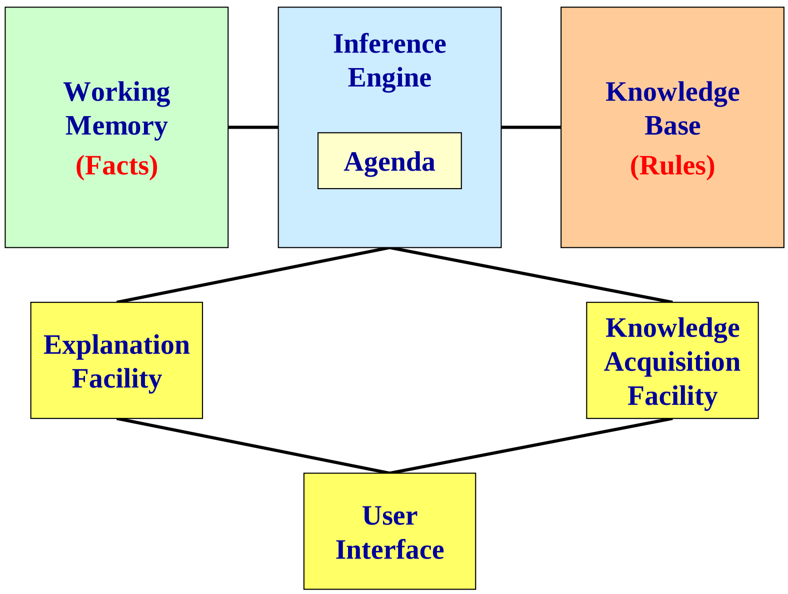

Figure 30: Structure of a rule-based expert system [Giarratano, Riley, 1998]

Table 13: Components of a rule-based expert system [Giarratano, Riley, 1998]

Knowledge base	Contains the domain knowledge needed to solve problems. In a rule-based system, knowledge is coded in the form of rules. The knowledge base is also called the production memory in a rule-based expert system.
Working memory	A global database of facts used by the rules.
Inference engine	Makes inferences by deciding which rules are satisfied by facts or objects, prioritizing the satisfied rules, and executing the rule with the highest priority.
Agenda	A prioritized list of rules created by the inference engine, whose patterns are satisfied by facts or objects in working memory.
Explanation facility	Explains the reasoning of the system to a user.

Knowledge acquisition facility	An automatic way for the user to enter knowledge in the system instead of having the knowledge engineer explicitly codes the knowledge.
User interface	The mechanism by which the primary user(s) and the expert system communicate.

14.3.1 Knowledge Base

The knowledge base (KB) is the organized repository for the collection of knowledge related to a domain and used for understanding, formulating, and solving problems in the knowledge-based system [Stefik, 1995]. The basic elements that are usually included in the KB are [Turban, Aronson, 1998]:

- facts, such as the problem situation and theory of the problem area,
- special heuristics, rules and hints that direct the use of knowledge to solve specific problems in a particular domain, and
- global strategies, which can be both heuristics and a part of the theory of the problem area.

Informally, a KB is a set of representations of facts about the world [Russell, Norvig, 1995]. The declarative approach to system building expresses knowledge in the form of sentences. This simplifies the construction problem enormously. Knowledge is contained in computer systems in the form of sentences that are stored in a KB in a knowledge representation language. There must be a way to add new sentences to the KB, and a way to query what is known.

Once a KB is built, AI techniques are used to give the computer an inference capability based on the facts and relationships contained in the KB [Turban, Aronson, 1998]. That is, the KB contains a data structure that can be manipulated by an inference system that uses search and pattern matching techniques on the KB to answer questions, draw conclusions, or otherwise perform an intelligent function. With a KB and the ability to draw inferences from it, the computer can be put to practical use as a problem solver and/or decision assistant. By searching the KB for relevant facts and relationships, the computer can reach one or more alternative solutions to the given problem, in support to the decision maker(s). The KB can be organized in several different configurations to facilitate fast inference (or reasoning) from the knowledge [Turban, Aronson, 1998]. Finally, the knowledge in the KB may be organized differently from that in the inference engine.

14.3.2 Working Memory (Workplace)

The working memory is a global database of facts used by the rules [Giarratano, Riley, 1998]. There are differences between the knowledge base and the working memory. For example, facts do not interact with one another.

The workplace is an area of working memory set aside for the description of a current problem, as specified by the input data; it is also used for recording intermediate results [Turban, Aronson, 1998]. The workplace records intermediate hypotheses and decisions. Three types of decisions can be recorded on the workplace [Turban, Aronson, 1998]:

- A plan (how to attack the problem)
- An agenda (potential actions awaiting execution)
- A solution (candidate hypotheses and alternative courses of action that the system has generated thus far)

14.3.3 Inference Engine

The brain of the expert system is the inference engine, also known as the control structure or rule interpreter (in rule-based expert systems) [Turban, Aronson, 1998]. This component is essentially a computer program that provides a methodology for reasoning about information in the knowledge base and in the working space, and for formulating conclusions. This component provides directions about how to use the system's knowledge by developing the agenda that organizes and controls the steps taken to solve problems whenever consultation is performed. The inference engine has three major components [Turban, Aronson, 1998]:

- An interpreter (rule interpreter in most systems), which executes the chosen agenda items by applying the corresponding knowledge base rules.
- A scheduler, which maintain control over the agenda. It estimates the effects of applying inference rules in light of item priorities or other criteria on the agenda.
- A consistency enforcer, which attempts to maintain a consistent representation of the emerging solution.

14.3.4 Expert Interface

The expert interface, not shown in Fig. 30, is the interface by which knowledge is entered into the system [Stefik, 1995]. It is used by a knowledge acquisition team consisting of one or more experts and a knowledge engineer. It provides access for updating, testing, and debugging a knowledge base, including tools for examining the contents of the knowledge base. By implication, not all users can update a system's knowledge base.

14.3.5 Knowledge Acquisition Facility

The knowledge acquisition facility is an optional feature on many systems [Giarratano, Riley, 1998]. In some expert system tools, the tool can learn by rule induction through examples and can automatically generate rules. This being said, general rules constructed by a knowledge engineer can be much more complex than the simple rules from rule induction.

14.3.6 Explanation Facility (Justifier)

A major feature of most expert systems is their ability to explain their advices or recommendations [Turban, Aronson, 1998]. Human experts are often asked to explain their views, recommendations, or decisions. If expert systems are to mimic humans in performing highly specialized tasks, they need to justify and explain their actions as well. An explanation is an attempt by an expert system to clarify its reasoning, recommendations, or other actions (such as asking questions). The part of an expert system that provides explanations and justification is called the justifier, or explanation facility/subsystem. It enables the system to examine its own reasoning and explain its operation. An explanation facility is an integral part of sophisticated expert systems [Giarratano, Riley, 1998].

The explanation facility has several purposes [Turban, Aronson, 1998]:

- Make the system more intelligible to the user.
- Uncover the shortcomings of the rules and knowledge base (debugging the systems by the knowledge engineer).
- Explain situations that were unanticipated by the user.
- Satisfy psychological and social needs by helping a user feel more assured about the actions of the expert system.
- Clarify the assumptions underlying the system's operations, to both the user and the builder.
- Conduct sensitivity analyses (using the explanation facility as a guide, the user can predict and test the effects of changes on the system).

Explanation is an extremely important function. One reason that an explanation facility is very important is that human life and property may depend on the answers of the expert system [Giarratano, Riley, 1998]. Because of the great potential for harm, an expert system must be able to justify its conclusions in the same way a human expert can explain why a certain conclusion was reached. Thus, an explanation facility provides an understandable check of the reasoning for humans.

A second reason for having an explanation facility occurs in the development phase of an expert system to confirm that the knowledge has been correctly acquired and is being correctly used by the system [Giarratano, Riley, 1998]. Because the expert system relies on inference, it must be able to explain its reasoning so that its reasoning can be checked. This is important in debugging because the knowledge may be incorrectly entered by typos or be incorrect due to misunderstandings between the knowledge engineer and the expert. A good explanation facility allows the expert and the knowledge engineer to verify the correctness of the knowledge.

In developing large expert systems, the need for a good explanation facility is essential [Turban, Aronson, 1998]. Large expert systems always include more facts and rules than one can easily remember. Often, a new rule added during expert system development will interact with other rules and data in unanticipated ways (and could make the expert system display strange explanations). An additional source of error may be unforeseen interactions in the expert system, which may be detected by running test cases with known reasoning that the system should follow [Giarratano, Riley, 1998].

As can be figured out from the discussion above, the ability to trace responsibility for conclusions to their sources is crucial both in the transfer of expertise and in problem solving. The explanation subsystem can trace such responsibility and explain the expert system behaviour by interactively answering questions such as the following [Turban, Aronson, 1998]:

- Why was a certain question asked by the expert system?
- How was a certain conclusion reached?
- Why was a certain alternative rejected?
- What is the plan to reach the solution? (for example, what remains to be established before a final diagnosis can be determined?)

Constructing explanations can become a very complex task, especially when done by machines [Turban, Aronson, 1998]. For this reason, many of the explanation facilities available in the development tools provide only two basic types of explanation: the why and the how. A typical why question is posed by the user to the computer after the computer asks the user to provide some information. The typical how question is posed by users when they would like to know how a certain conclusion or recommendation was reached. Simple systems are limited to the final conclusion. More complex systems can handle intermediate conclusions as well. The system can explain why a certain rule was fired, i.e., it shows the chain of rules used to reach the conclusion. The why and how explanations often show the rules as they were programmed and not in a natural language.

A typology of expert system explanations includes [Turban, Aronson, 1998]:

- <u>Trace, or line of reasoning</u>, which refers to a record of the inferential steps taken by an expert system to reach a conclusion.
- <u>Justification</u>, which is an explicit description of the causal argument or rationale behind each inferential step taken by the expert system (based on the empirical associations involving the encoding of large chunks of knowledge).
- <u>Strategy</u>, which is a high-level goal structure that determines how the expert system uses its domain knowledge to accomplish a task (or metaknowledge)

Depending on the system, an explanation facility may be simple or elaborate [Giarratano, Riley, 1998]. A simple explanation facility in a rule-based system may list all the facts that made the latest rule execute. Explanation in rule-based expert systems is usually associated with some way of tracing the rules that are fired during the course of a problem-solving session [Turban, Aronson, 1998]. This is about the closest to a real explanation that today's systems come, given that their knowledge is usually represented almost exclusively as rules that do not include basic principles necessary for a human-type explanation [Turban, Aronson, 1998].

There are special problems in explaining the reasoning process when rules are (automatically) induced [Turban, Aronson, 1998]. Such systems can generally only display the rules and leave the real explanation up to the user. Explanation in non-rule-based systems is much more difficult than in rule-based ones because the inference procedures are more complex.

Justification requires a deeper understanding of the domain than the trace method [Turban, Aronson, 1998]. By demonstrating that the conclusions developed by the system are based on sound reasoning, justification increases expert system users' confidence in the problem-solving ability and hence the acceptability of the conclusions. Because expert systems can achieve high performance levels only within narrow problem areas, justification enables users to make more informed decisions on whether the advice is to be followed. Expert system explanation facilities can make a system's advice more acceptable to users. Justification is the most effective type of expert system explanation to bring about change in users' attitude toward the system.

Strategy involves knowledge about the problem solving procedure [Turban, Aronson, 1998]. Generally, the last two explanation types are more difficult to perform in an expert system because strategic knowledge tends to be buried implicitly in the knowledge base, and an expert system does not need an explicit representation of justification knowledge to execute.

Elaborate explanation facilities may be designed to allow the user to explore multiple lines of “what if” or hypothetical reasoning questions and even to translate natural language into rules [Giarratano, Riley, 1998]. Elaborate systems may do what is listed in Table 14.

Table 14: Features of an elaborated explanation facility [Giarratano, Riley, 1998]

List all the reasons for and against a particular hypothesis.	A hypothesis is a goal that is to be proved. In a real problem there may be multiple hypotheses. A hypothesis can also be viewed as a fact whose truth is in doubt and must be proved.
List all the hypotheses that may explain the observed evidence.	
Explain all the consequences of a hypothesis.	
Give a prognosis or prediction of what will occur if the hypothesis is true.	
Justify the questions that the program asks of the user for further information.	These questions may be used to direct the line of reasoning. In most real problems it is too expensive or takes too long to explore all possibilities, and some way must be provided to guide the search for the correct solution.
Justify the knowledge of the program.	

A hypothesis is justified by knowledge, and the knowledge is justified by a warrant that it is correct [Giarratano, Riley, 1998]. A warrant is essentially a meta-explanation that explains the expert system's explanation of its reasoning.

There are different methods for generating explanations [Turban, Aronson, 1998]. One easy way is to pre-insert pieces of English text (scripts) in the system. For example, each question that could be asked by the user may have an answer text associated with it. This is called static explanation. Several problems are associated with static explanations. For example, all questions and answers must be anticipated in advance. For large systems, this is very difficult. The system also has essentially no idea about what it is saying. In the long run, the program may be modified without changing the text, thus causing inconsistency. A better form of explanation is a dynamic explanation, which is reconstructed according to the execution pattern of the rules. In this method, the system reconstructs the reasons for its actions as it evaluates rules.

Most expert system explanation facilities consist of printing out a trace of the rules being used [Turban, Aronson, 1998]. Explanation is not treated as a task that requires intelligence in itself. If expert systems are to provide satisfactory explanations, future systems must include not only knowledge of how to solve problems in their respective domains, but also knowledge of how to effectively communicate to users their understanding of this problem-solving process. Obviously, the balance of these two types of knowledge varies according to the primary function of the system. Constructing such knowledge bases will involve formalizing the heuristics used in providing good explanations. With current expert systems, much of the knowledge vital to providing a good explanation (such as knowledge about the system's problem-solving strategy) is not expressed explicitly in rules. Therefore, the purely rule-based representation may be difficult to grasp, especially when the relationships between the rules are not made explicit in the explanation.

The system's knowledge about how it reasons is called metaknowledge, or knowledge about knowledge [Turban, Aronson, 1998]. Metaknowledge allows the system to examine the operation of the declarative and procedural knowledge in the knowledge base. Explanation can be viewed as another aspect of metaknowledge. Over time, metaknowledge will allow expert systems to do even more. They will be able to create the rationale behind individual rules by reasoning from first principles. They will tailor their explanations to fit the requirements of their audience. And they will be able to change their own internal structure through rule correction, reorganization of the knowledge base, and system reconfiguration.

14.4 Execution Flow of Rule-Based Expert Systems

Multiple rules may apply to a given situation about which the expert system is reasoning [Giarratano, Riley, 1998]. The flow of execution is not sequential in an expert system so that one cannot just read its code line by line and understand how the system operates. That is, the order in which rules have been entered in the system is not necessarily the order in which they will be executed. The expert system acts much like a parallel program in which the rules are independent knowledge processor.

14.5 Consultation Mode of Expert Systems

Once the expert system is developed and validated, it is deployed to the users [Turban, Aronson, 1998]. When users seek advice, they access the expert system. The expert system conducts a bidirectional dialog with the user, asking the user to provide facts about a specific situation. While accepting the user's answers, the expert system attempts to reach a conclusion. This effort is made by the inference engine, which chooses search techniques to be used to determine how the rules in the knowledge base are to be applied to the problem. The user can ask for explanations. The quality of the inference capability is determined by the quality and completeness of the rules (or appropriateness and depth of the knowledge representation), by the knowledge representation method used, and by the power of the inference engine.

14.6 Types of Expert Systems

Expert systems appear in many varieties [Turban, Aronson, 1998]. Many commercial expert systems are *rule-based systems*. In such systems the knowledge is represented as a series of production rules. In *framed-based systems*, the knowledge is represented as frames, a representation of the object-oriented approach. *Hybrid systems* include several knowledge representation approaches. *Model-based systems* are structured around a model that simulates the structure and function of the system under study. The model is used to compute values, which are compared to observed ones. The comparison triggers action (if needed) or further diagnosis. *Real-time expert systems* have a strict limit on the system's response time, which must be fast enough to control the process being computerized. In other words, the system always produces a response by the time it is needed.

14.7 Desirable Characteristics of an Expert System

An expert system is usually designed to have the desirable characteristics listed in Table 15.

Table 15: Desirable characteristics of an expert system [Giarratano, Riley, 1998]

High performance	The system must be capable of responding at a level of competency equal or better than that of an expert in the field. That is, the quality of the advice given by the system must be very high.
Adequate response time	The system must perform in a reasonable amount of time, comparable to or better than the time required by an expert to reach a decision. An expert system that takes a year to reach a decision compared to an expert's time of one hour would not be too useful. The time constraints placed on the performance of an expert system may be especially severe in the case of real-time systems, when a response must be made within a certain time interval.
Good reliability	The expert system must be reliable and not prone to crashes or it will not be used.
Understandable	The system should be able to explain the steps of its reasoning while executing so that it is understandable. Rather than being just a “black box” that produces a miraculous answer, the system should have an explanation capability in the same way that human experts can explain their reasoning. This feature is very important for several reasons.
Flexibility	Because of the large amount of knowledge that an expert system may have, it is important to have an efficient mechanism for adding, changing, and deleting knowledge. One reason for the popularity of rule-based systems is the efficient and modular storage capability of rules.

14.8 Conventional Systems Versus Expert Systems

Table 16 provides a comparison of conventional systems and expert systems.

Table 16: Comparison of conventional systems and expert systems [Turban, Aronson, 1998]

Conventional Systems	Expert Systems
The information and its processing are usually combined in one sequential program.	The knowledge base is clearly separated from the processing (inference) mechanism (e.g., knowledge rules are separated from the control)
The program does not make mistakes (programmers do)	Program may make mistakes.
Do not (usually) explain why input data are needed or how conclusions are drawn.	Explanation is part of most expert systems.
Require all input data. May not function properly with missing data, unless planned for.	Do not require all initial facts. Typically can arrive at reasonable conclusions with missing facts.
Changes in the program are tedious.	Changes in the rules are easy to make.
The system operates only when it is completed.	The system can operate with only a few rules (as the first prototype).
Execution is done on a step-by-step (algorithmic) basis.	Execution is done by using heuristics and logic.
Effective manipulation of large databases.	Effective manipulation of large knowledge bases.
Representation and use of data.	Representation and use of knowledge.
Efficiency is a major goal.	Effectiveness is the major goal.
Easily deal with quantitative data.	Easily deal with qualitative data.
Use numerical data representation.	Use symbolic knowledge representation.
Capture, magnify, and distribute access to numeric data or information.	Capture, magnify, and distribute access to judgement and knowledge.

14.9 Benefits of Knowledge-Based Expert Systems

The process of building knowledge systems, like the process of scientific discovery, is arguably a process of invention [Stefik, 1995]. Terms need to be established, representations and models need to be created, questions and processes need to be designed and tested. Most knowledge-system projects are not started with the goal of advancing the state of knowledge. Rather, they are intended for mechanizing what is thought to be rather mundane and well-understood bodies of knowledge. Nonetheless, it often turns out that the requisite formalization and systematic testing leads to increased understanding in the domain. The crucial creative act in designing a knowledge system is to characterize the problem as a systematic and practical search of space of possibilities. This creative step itself can sometimes be a contribution to the science of the domain, and it often exposes some elements of the domain that have escaped systematic and careful attention.

In addition to the valuable increased understanding of domain knowledge discussed above, expert systems have a number of additional attractive features. Some benefits of expert systems are listed in Table 17.

Table 17: Some benefits of expert systems ([Turban, Aronson, 1998] and [Giarratano, Riley, 1998])

Capture of scarce expertise.	The scarcity of expertise becomes evident in situations where there are not enough experts for a task, the expert is about to retire or leave a job, or expertise is required over a broad geographic area.
Permanence.	The expertise is permanent. Unlike human experts, who may retire, quit, or die, the expert system's knowledge will last indefinitely.
Increased availability/ accessibility to knowledge.	Expert systems make knowledge, such as scarce expertise, and information accessible, thus freeing experts from routine work. People can query systems and receive advice. Expert systems can also achieve knowledge transfer to remote locations. Expertise is available on any suitable computer hardware. In a very real sense, an expert system is the mass production of expertise.
Improved decision-making process	Expert systems enhance decision making and problem solving by allowing the integration of top expert's judgment into the analysis.
Improved decision/ product quality.	Expert systems enable a consistency of performance, i.e., they consistently pay attention to all details and do not overlook relevant information and potential solutions, thereby reducing the size and rate of errors, and ensuring a standard high quality of the results
Increased reliability.	Expert systems are reliable. They do not become tired or bored, call in sick, or go on strike, and they do not talk back to the boss.
Steady, unemotional, and complete response at all time.	This may be very important in real-time and emergency situations, when a human expert may not operate at peak efficiency because of stress or fatigue.
Multiple expertises.	In certain cases, expert systems may force one to integrate the opinion of several experts. The knowledge of multiple experts can thus be made available to work simultaneously and continuously on a problem at any time of day or night. The level of expertise combined from several experts may exceed that of a single human expert (i.e., the expert system may even be more accurate than any single expert).
Increased communication among experts.	Expert systems may facilitate communication among decision makers in a team.
Increased confidence.	Expert systems increase confidence that the correct decision was made by providing a second opinion to a human expert or break a tie

	in case of disagreements by multiple human experts. Of course, this method probably won't work if the expert system was programmed by one of the experts. The expert system should always agree with the expert, unless a mistake was made by the expert. However, this may happen if the human expert was tired or under stress. The expert system can also explicitly explain in detail the reasoning that led to a conclusion. A human may be too tired, unwilling, or unable to do this all the time. This also increases the confidence that the correct decision is made.
Fast response / Decreased decision-making time.	Fast or real-time response may be necessary for some applications. Using the system's recommendations, a human can make much faster decisions. Depending on the software and hardware used, an expert system may respond faster and be more available than a human expert. Some emergency situations may require responses faster than a human and so a real-time expert system is a good choice.
Better understanding of the decision-making situation.	Expert systems can provide rapid feedback on decision consequences, thus providing a better understanding of the decision-making situation.
Ability to solve complex problems.	Expert systems may solve problems whose complexity exceeds human ability. Expert systems are able to solve problems where the required scope of knowledge exceeds that of any one individual. This allows decision makers to gain control over complicated situations and ease the operation of complex systems.
Ability to work with incomplete or uncertain information.	In contrast to conventional computer systems, expert systems can, like human experts, work with incomplete, imprecise, uncertain data, information, or knowledge. The user can respond with a “don't know” or “not sure” answer to one or more of the system's questions during a consultation, and the expert system will still be able to produce an answer, although it may not be a certain one.
Improved performance in dynamic, unpredictable environments.	Expert systems allow rapid response to unforeseen changes in the environment.
Reduced danger for operations in hazardous environments.	Many tasks require humans to operate in hazardous environments. Expert systems may enable humans to avoid such environments. This characteristic can be very important in military conflicts.
Increased capabilities of other computerized systems.	Integration of expert systems with other systems makes the other systems more effective: they cover more applications, work faster, and produce higher-quality results.

Provide training.	Novices who work with expert systems become more and more experienced. The explanation facility can also serve as a teaching device, and so can notes and explanations that may be inserted in the knowledge base.
Flexibility.	Expert systems can offer flexibility.
Increased output and productivity.	As expert systems may work faster than humans, increased output and productivity is possible.
Reduced cost.	The cost of providing expertise per user is greatly lowered.

14.10 Limitations of Knowledge-Based Expert Systems

In whatever role one employs expert systems, those systems require knowledge to be competent [Stefik, 1995]. A key insight is that the power of an expert system is derived from the specific knowledge it possesses, not from the particular formalisms and inference schemes it uses [Turban, Aronson, 1998]. Problem-solving techniques with wide or universal applicability are referred to as "weak" methods; the reason is that their universality typically makes them less effective than approaches that have been carefully tailored to solve specific problems [Ginsberg, 1993]. Many fielded AI systems use special-purpose approaches that, although useful in only narrow domains, have had significant practical impact in these domains.

This being said, one might ask whether it is the case that all expert systems use only a little bit of knowledge [Stefik, 1995]. If one compares the amount of knowledge in knowledge-based expert systems with the amount that most people have, the answer to this is unmistakably yes. Knowledge-based systems typically fall short of human expertise, at least in breadth. They know less than a young child, arguably less than most household pets.

A problem with expert systems is thus that their expertise is limited to the knowledge domain that the system knows about [Giarratano, Riley, 1998]. Even in the most successful applications where expert systems outperform human experts in their reliability and consistency of results, expert systems have less breadth and flexibility than human experts [Stefik, 1995]. They function well only in their restricted knowledge domains [Giarratano, Riley, 1998]. Typical expert systems cannot generalize their knowledge by using analogy to reason about new situations the way people can. This is one reason why a general-purpose problem solver still eludes the AI community. However, because of the nature of the tasks to which knowledge-based systems are applied and the extensive knowledge formulation processes that are used, expert systems tend to have much more special case knowledge about solving problems than most conventional software [Stefik, 1995].

Beyond the quantitative issue, there is also a qualitative issue [Stefik, 1995]. In the context of computers, some people find the term knowledge jarring. To paraphrase a typical complaint, "We may not be able to say exactly what knowledge is, but whatever it is, it is not the same as what is programmed in knowledge-based systems". However, this attitude seldom arises in the context of concrete examples of knowledge-based systems. When a knowledge-based system is being used or demonstrated, the situation draws one to ask specific questions about the particulars of its problem-solving knowledge rather than general questions about the nature of a machine's ability to know. In other words, one does not challenge whether the system carries knowledge.

One can also ask whether expert systems only work in areas where "stupidity" works [Stefik, 1995]. Given that all computer systems built to date are stupid when compared with people, this is correct. More specifically, knowledge-based systems and computer systems in general are often criticized as being too fragile, inflexible, and narrow. They violate one's expectations of what communicating with an intelligent or knowledgeable agent should be like. When expert systems make mistakes, these mistakes can appear quite ridiculous in human terms. Many simple concepts widely regarded as common sense are simply beyond them.

The considerations above lead to the following research goals [Stefik, 1995], i.e., understanding how to build knowledge-based systems that:

- do not break down or fail on slight variations of problems just outside their area of expertise;
- can reuse knowledge in similar, new situations;
- can flexibly integrate multiple kinds of knowledge and goals.

One idea is to try to build systems that can use very large knowledge bases in which many things could be found to apply in different situations. This has been called the breadth hypothesis, i.e., believing that a broad knowledge base would provide the basis for metaphorical and analogical reasoning.

Finally, some other limitations of expert systems are mentioned in [Turban, Aronson, 1998]:

- Users of expert systems have natural cognitive limits
- Lack of trust by end-users may be a barrier to the use of an expert system.
- Expert systems may not be able to arrive at conclusions (especially in early stages of system development).
- Expert systems, like human experts, sometimes produce incorrect recommendations.

14.11 Suitable Application Domains for Expert Systems

Before starting to build an expert system, it is essential to decide if an expert system is the appropriate paradigm [Giarratano, Riley, 1998]. For example, one concern is whether an expert system should be used instead of an alternative paradigm, such as conventional programming. Conventional computer programs are used to solve many types of problems. As previously discussed, these problems generally have algorithmic solutions that lend themselves well to conventional programs and programming languages. By contrast, expert systems are primarily designed for symbolic reasoning.

The appropriate domain for an expert system depends on a number of factors. Table 18 lists some of these factors.

Table 18: Some factors to identify the appropriate domain for an expert system [Giarratano, Riley, 1998]

Can the problem be solved by conventional programming?	If the answer is yes, then an expert system is not the best choice. Expert systems are best suited for situations in which there is no efficient algorithmic solution. Such cases are called ill-structured problems and reasoning may offer the only hope of a good solution.
Is the domain well bounded?	It is important to have well-defined limits on what the expert system is expected to know and what its capabilities should be. The points: when does one stop adding domains? The more domains, the more complex the expert system becomes.
Are there a need and a desire for an expert system?	Although it's great experience to build an expert system, it's rather pointless if no one is willing to use it.
Is there at least one human expert who is willing to cooperate?	There must be an expert who is willing, and preferably enthusiastic about the project. This being said, it might be wise to limit the number of experts involved. Different experts may have different ways of solving a problem. Sometimes they may even reach different conclusions. Trying to code multiple methods of problem solving in one knowledge base may create internal conflicts and incompatibilities.
Can the expert explain the knowledge so that is understandable by the knowledge engineer?	Even if the expert is willing to cooperate, there may be difficulty in expressing the knowledge in explicit terms. Typically, the knowledge engineer doesn't know the technical terms of the expert. It may take a long time for the knowledge engineer to even understand what the expert is talking about, let alone translate that knowledge into explicit computer code.
Is the problem-solving knowledge mainly heuristic and uncertain?	Expert systems are appropriate when the expert's knowledge is largely heuristic and uncertain. That is, the knowledge may be based on experience, called experiential knowledge, and the expert may have tried various approaches in case one doesn't work. In other words, the expert's knowledge may be a trial-and-error approach, rather than one based on logic and algorithms. However, the expert can still solve the problem faster than someone who is not an expert.

Table 19 lists a number of general problem areas or categories of interest for the use of expert systems.

Table 19: General problem categories for expert systems [Turban, Aronson, 1998]

Problem Category	Problem Addressed
Interpretation	Inferring situation description from observations. This includes surveillance, speech understanding, image analysis, signal

	interpretation, and many kinds of intelligence analysis.
Prediction	Inferring likely consequences of given situations.
Planning	Developing plans to achieve goals.
Monitoring	Comparing observations to plans, flagging exceptions.
Control	Interpreting, predicting, repairing, and monitoring system behaviours.

14.11.1 Ill-structured problems and Opportunistic Processing

Because there are typically so many possibilities, an ill-structured problem would not lend itself well to an algorithmic solution [Giarratano, Riley, 1998]. As stated in Table 18, expert systems are best suited for such problems. However, in dealing with ill-structured problems, there is a danger that the expert system design may accidentally mirror an algorithmic solution, i.e., the development of an expert system may unknowingly discover an algorithmic solution. An indication that this has happened is if a solution that requires a rigid control structure is found, i.e., the rules are forced to execute in a certain sequence by the knowledge engineer explicitly setting the priorities of many rules. Forcing a rigid control structure on the expert system cancels a major advantage of expert system technology, which is dealing with unexpected input that does not follow a predetermined pattern. That is, expert systems react opportunistically to their input, whatever it is. Conventional programs generally expect input to follow a certain sequence.

14.11.2 Dealing with Uncertainty

Expert systems are often designed to deal with uncertainty because reasoning is one of the best tools that has been discovered for doing so [Giarratano, Riley, 1998]. The uncertainty may arise in the input data to the expert system and even in the knowledge base itself. At first this may seem surprising to people used to conventional programming. However, much of the human knowledge is heuristic, which means that it may only work correctly part of the time. In addition, the input data may be incorrect, incomplete, or inconsistent, or have other errors. Algorithmic solutions are not capable of dealing with such situations because an algorithm guarantees the solution of a problem in a finite series of steps.

14.11.3 Expert Systems as Assistants to Experts

Expert systems can be used by human experts as knowledgeable, intelligent assistants [Turban, Aronson, 1998]. As more knowledge is added to the intelligent assistant, it acts more like an expert [Giarratano, Riley, 1998]. Thus, developing an intelligent assistant may be a useful milestone in producing a complete expert system. In addition, it may free up more of the expert's time by speeding up the solution of problems.

14.11.4 Other Applications

Knowledge-based expert systems can also be used [Giarratano, Riley, 1998] [Stefik, 1995] [Turban, Aronson, 1998]:

- by non-experts to improve their problem-solving capabilities;
- as an intelligent tutor by letting a student run sample programs and by explaining the system's reasoning;
- to access a database in an intelligent manner;
- to solve routine problems or routine parts of problems so the user can focus on harder and more interesting problems;
- to check work;
- to extend methods to larger problems than one can solve by hand;
- to work systematically on large cases, relying on the tirelessness of an automatic system;
- as a computational and experimental medium for expressing, testing, and extending a theory.

14.12 Expert Systems Success Factors

Many researchers have investigated the reasons why expert systems succeed and fail in practice. Two of the most critical factors are the champion in management and user involvement and training. Some other success factors for expert systems are [Turban, Aronson, 1998]:

- The level of knowledge must be sufficiently high.
- Expertise must be available from at least one cooperative expert.
- The problem to be solved must be
    - qualitative (fuzzy), not quantitative (otherwise, use a numerical approach).
    - important and difficult enough to warrant development of an expert system (but it need not be a core function).
    - sufficiently narrow in scope.
- The expert system shell characteristics are important. It must be of high quality, and naturally store and manipulate the knowledge.
- The user interface must be friendly for novice users.
- Knowledgeable system developers with good people skills are needed.
- The impact of expert systems as a source of end-users' job improvement must be considered. The impact should be favourable. End-user attitudes and expectations must be considered.
- Management support must be cultivated.

15. Some Critical Research Issues in SAIF

Situation analysis and information fusion (SAIF) processes involve both people and machines. Three distinct types of processes are involved [Lambert, 2003]: psychological processes associated with people; technological processes characteristic of machines; and integration processes facilitating interaction between the psychological and technological processes. In this framework, SAIF draw together concepts from a wide range of diverse fields: psychology, human factors, knowledge representation, artificial intelligence, mathematical logic and signal processing, just to name a few. Despite the importance place around SAIF in all DND strategic documents, the current supporting systems and the associated technology being exploited often fail to meet the demanding requirements of the decision makers, and major S&T advancements in all these fields are required to achieve the full potential of the related enabling technologies and to best serve the operational military and national security communities.

Among the important issues requiring advancements is knowledge modeling and representation, as situation awareness is usually associated with having perception, comprehension and projection knowledge about the world. Investigating means of supporting situation analysis requires acquiring a deep understanding of the ways humans represent the world, and the development and exploitation of efficient machine representations of relevant aspects of the world. This will be a lot less trivial in the future, as one will have to describe the threats and perform information fusion in the three domains of human reality. These are [Waltz, 2004]:

- the physical domain (e.g., mass and energy, human bodies, infrastructures, etc.),
- the cognitive domain (e.g., emotion, intention, etc.), and,
- the symbolic domain (e.g., explicit information, media content, etc.).

Formal methods of ontological engineering can be applied with the goal of producing defensible representations of the world (e.g., situational constructs). The ambition is not to capture “the” meaning of the relevant aspects of the world, but to formulate “a” meaning of these aspects that is sufficient to engineer SAIF systems [Lambert, 2003]. One goal is to ensure that inconsistencies and ignorance in a SAIF system derive from the system’s knowledge of the world, and not from the meaning of the terms by which the system describes the world. The semantic challenge discussed in [Nowak, Lambert, 2005] is linked to the development of ontologies appropriate for SAIF.

There is a need to distinguish *the entire world* (i.e., all aspects of the world) from *the world of interest* to some decision maker (i.e., only aspects of interest). The spectrum of world states of interest as associated with a given task or mission is bounded by, and related to, the informational needs of users [Llinas et al, 2004]. In this regard, work domain models can be constructed based on cognitive engineering techniques in order to understand and formally document the information needs of decision makers.

On one hand, ontological engineering can be applied to generate representations of the world, while, on the other hand, cognitive engineering techniques can be used to capture the informational needs of users. Clearly, one could say that the cognitive engineering techniques used for developing a specification of user information needs could inform the ontologist in developing a correspondingly-bounded ontological description of those needs as reflected in a set of ontologically-described states [Llinas et al, 2004]. Concepts identified in the ontological modeling could be mapped onto concepts identified in the work domain modeling to insure consistency across and within the two modeling constructs. There would result from this exercise an ontological specification of the states of interest couched in an ontological representation language.

Just as important as modeling and representing knowledge is modeling, representing and managing uncertainty. Situation analysis is all about reasoning under uncertainty, and the essence of information fusion is to reduce uncertainty about aspects of the world of interest. The data and information sources must be characterized with respect to the uncertainty attached to their outputs, mathematical techniques are required to manage it, and computer display representations are ultimately required to communicate it to the decision makers, as uncertainty can hopefully be reduced through the use of fusion techniques, but it can never be totally eliminated. Uncertainty is a widely used term within the artificial intelligence and engineering communities. However, the authors in these fields of application and research do not always agree on the meaning of uncertainty, on its different types, on the possible sources, on the synonyms, on possible classifications, on representations, etc. [Jousselme, Maupin, Bossé, 2003]. There is a need to better explore the concept of uncertainty and its related concepts such as imperfection, imprecision, vagueness, ambiguity, incompleteness, ignorance, etc.

Better exploitation of contextual knowledge is necessary for situation analysis. One must describe, understand and consider contextual knowledge of the threats, i.e., fusion must describe and reason about contextual information and situations [Waltz, 2004]. Context establishes the basis for discerning meaning of its subjects. It may occur at many levels. Contextual knowledge provides a holistic view, and insight into the subjects. [Powell et al, 2006] discuss the role of context in the interpretation of complex battlespace intelligence.

With respect to the data fusion domain, context has to do with what is typically referred to as higher levels of fusion. In this regard, there is a growing need to extend well beyond the traditional machine sensor fusion emphasis of the data fusion community, by including higher-level information fusion considerations involving both humans and machines [Lambert, 2003]. This requires a more strategic foundation.

Situation projection must produce an estimate of future possibilities for situation elements, based on current trends. This situation analysis capability is necessary because one is not only concerned with what is happening, but also with what events and/or activities are going to happen next. The decision maker can never influence the present, only the future. Hence, knowledge of the current world state is only of value as a contribution to understanding the future. Situation element projection must produce an estimate of future possibilities for situation elements, based on current trends and expectations. Ultimately, the predictive capability can include story building, simulation, war gaming, engagement modeling, etc.

The inference and situation projection processes are both very challenging issues, and they will represent even bigger challenges when addressing the hard (difficult, intractable) targets threats of the 21st century for which one will have to understand the behaviour of threats as complex adaptive systems, i.e., consider targets not as entities, but as highly interacting sets of independent entities in complex adaptive systems. There is a need to evolve from “traditional” *complicated* systems toward *complex* systems [Waltz, 2004]. For the former, there is some linearity and one tries to estimate the hidden order, considering things such as weapon acquisition, order-of-battle analysis, communication network mapping, mobility analysis, etc. For complex systems, non-linear and un-ordered, one needs to care about group dynamics and decision making, perceptions and intentions, adaptive strategies, etc. Consequently, the fusion process must model the dynamics of non-linear systems in such situations [Waltz, 2004].

15.1 SAIF Domain and Peripheral Concepts and Issues

Figure 31 presents a holistic R&D perspective of situation analysis and information fusion, where all important and necessary aspects are listed. It represents some sort of “checklist” that could be used as a guideline for the formulation of a complete, long-term R&D program to support the development of entire SAIF systems.

Must understand	Must apply/contribute to	Must take into account
• Decision Making (DM) • Situation AWareness (SAW) – Knowledge ▸ Mental representation (a state) • Situation Analysis (SA) – Mental process – Perceive, comprehend, project • Epistemology • Psychology • Military & public security operations • Scenario(s) – Situation elements ▸ The environment – Goals • Metaphysics – Ontology • Complexity – Networks: physical, social, etc. – Emergent behaviours – Non-linearities • Sensing (the five senses) • Sensing/measurement/collection technologies – Signal Processing – Operating characteristics – Strengths and weaknesses – Dynamic vs "static" (a priori) • Mental inference / reasoning – Deduction, induction, abduction • Mental Fusion – Alignment, correlation, fusion • Human limitations – Uncertainty (mental) • Technological support – SAW requirements regarding tech. supp. • Shared situation awareness • Team decision making	• Knowledge Engineering – Acquisition/Capture/Elicitation ▸ Machine learning ▸ Knowledge discovery, data mining – Validation – Modeling & machine representation • SAW requirements – Cognitive engineering ▸ KID requirements ▸ Visual representation requirements • Ontological engineering – Develop paper & machine ontologies of ▸ The world ▸ Situation(s) & situation analysis ▸ The world of interest ▸ Military & PS ops. ▸ Support technology • Complexity/complex systems theory • Managing the specific to generic spectrum(s) • KID Sources characterization • Interfacing sources and SAIF engine • Artificial intelligence (reasoning/inference) – Problem solver – Expert systems – Explanation & justification capability • Automated data & information fusion • Logic • Mathematics • Uncertainty management – Machine representation – Fusion/reasoning under uncertainty • SAIF resources management – Adaptive sensing / collection – Adaptive processing • SAIF processing architecture – Data models & structures – Proc. paradigm (proc., BB, multi-agents) • SAW measurement/quality assessment – Methodology & metrics & test scenarios • SAIF support systems & HC interaction – Level of automation	• KID storage/retrieval technologies – Database technologies – Data warehousing technologies • Human factors • Visualization technologies • Processing tree – KID distribution – Processing nodes • Information management – Communications – KID assurance, security – Interoperability • Systems-of-systems integration architecture – Tools/services integration architecture – Coalition architecture • Knowledge management technologies – Organize and apply knowledge assets – Create, reuse, leverage knowledge assets – Integrate, share knowledge assets – Knowledge discovery – KID retrieval, triage, packaging – Portfolio – Semantic web • Distributed collaboration technologies • COTS products

Figure 31: R&D perspective of situation analysis and information fusion

The elements of such a program have been sorted into three main categories, corresponding respectively to the aspects that the members of an R&D group:

- “must understand”. These are the underlying fundamental background elements of the SAIF domain. Typically, the members of the R&D group would not contribute to advancements in these fields per se, but they would beneficiate a lot from gaining a fair understanding of such issues. It would provide the group with some serious depth, thereby augmenting the R&D credibility of the group.

- "must apply / contribute to". These are the elements frequently manipulated by the SAIF experts of the SAIF community, and that are at the core of any SAIF system. The members of the R&D group should typically contribute to advancements in these fields and publish their results to make them available to their peers of the community.
- "must take into account". Clearly, a SAIF system will not (actually, it cannot) operate in isolation from the rest of the world. The necessary knowledge/information/data must be made available to the system, and the system requires the necessary means to make its output available to the user(s). The SAIF system will be integrated into a larger (command and control) system, as a "tool" or a "service" among many others. Typically, the members of the R&D group would not contribute to advancements in these fields per se; instead, such issues should be viewed as eventual constraints/enablers that the SAIF system designers will eventually have to care of.

Many aspects of Fig. 31 have been reviewed and examined in the previous sections. However, a complete discussion of all issues is clearly out of the scope of this report.

16. Conclusion

This report presented a new perspective of situation analysis and information fusion (SAIF). Mostly based on the fact that awareness ultimately has to do with having knowledge of something, the author proposed that developing and adopting a knowledge-centric view of situation analysis should provide a more holistic perspective of this process, leading to the development of better, more adequate SAIF support systems for the operational communities.

The report didn't present “solutions”, or findings and results of completed activities. The intent was more to provide a fair overview of many of the main issues around SAIF and knowledge-based systems, in order to set up the foundational R&D framework for two projects that are just starting under the Applied Research Program (ARP) at Defence R&D Canada: SATAC and CKE-4-MDA. Among other things, C2, intelligence, SAIF, support systems, knowledge, knowledge representation and management, reasoning processes, methods and systems, artificial intelligence, processing/computing/programming paradigms, and expert systems were discussed.

SAIF has a critical role to play in the C2 and intelligence processes. For this reason, numerous ongoing DND and DRDC projects have an important SAIF component. However, despite the importance place around SAIF in many DND strategic documents and projects, the current systems and the associated technology being exploited often fail to meet the demanding requirements of the operational decision makers; major science and technology advancements are required to really achieve the full potential of the related enabling technologies and to best serve the operational communities. The effort reported here is one step in this direction; it constitutes some solid basis on which a long-term R&D program should be built.

Clearly, adopting the knowledge-centric view of SAIF and developing a complete support system is a very challenging, multidisciplinary enterprise. Project teams will certainly have to seriously deepen the aspects described in this report along the way. Along this line of thought, R&D activities have been and are still currently being conducted to further investigate knowledge representation concepts, paradigms and techniques [Roy, Auger, 2007-A], and reasoning processes, methods and systems [Roy, Auger, 2007-B] for use in knowledge-based SAIF support systems. Adopting the knowledge-centric view of SAIF requires that knowledge (i.e., expertise) is eventually acquired from the subject matter experts (SMEs) of the different military and public security application domains. In this regard, knowledge and ontological engineering techniques have been and are still being investigated at the moment [Roy, Auger, 2007-C]. Knowledge acquisition and validation sessions with SMEs are about to be conducted. Finally, a number of critical research issues in SAIF (e.g., reasoning under uncertainty, exploitation of contextual knowledge, complexity theory) were briefly discussed in this report; these aspects need to be seriously studied if significant advancements are to be made in this field. Actually, other issues not even discussed here may also need to be eventually considered.

References

[Alberts et al, 2001], Alberts, D. S., Garstka, J. J., Hayes, R. E. and Signori, D. A., Understanding Information Age Warfare, CCRP Publication series, DOD Command and Control Research Program, 2001. http://www.dodccrp.org/

[Alberts, Hayes, 2003], Alberts, D. S. and Hayes, R. E., Power to the Edge, CCRP Publication series, DOD Command and Control Research Program, 2003. http://www.dodccrp.org/

[Alberts, Hayes, 2006], Alberts, D. S. and Hayes, R. E., Understanding Command and Control, CCRP Publication series, DOD Command and Control Research Program, 2006. http://www.dodccrp.org/.

[Antony, 1995], Antony, R. T., Principles of Data Fusion Automation, Artech House, Norwood, MA, 1995.

[APQC, 2000], American Productivity and Quality Center, Knowledge Management: A Guide for Your Journey to Best-Practice Processes, Houston: APQC, 2000.

[Boyd, 1987], Boyd, J. R., A Discourse on Winning and Loosing, August 1987.

[Bukowitz, Williams, 1999], Bukowitz, W. R. and Williams, R. L., The Knowledge Management Fieldbook, London: Pearson, 1999.

[Champoux, 1999], Champoux, P., Overview of the Knowledge Management Domain, DMR Consulting, Defence R&D Canada – Valcartier Contract Report, 1999.

[Dieng et al, 1998], Dieng, R., Corby, O., Giboin, A. and Ribière, M., Methods and Tools for Corporate Knowledge Management, Proceedings of KAW'98, Banff, Canada, 1998.

[DoD, 2006], DoD, Department of Defense Dictionary of Military and Associated Terms, Joint Pubs. 1-02 (As amended through 31 August 2005), May 2006. http://www.dtic.mil/doctrine/jel/doddict/

[Elm et al., 2002], Elm, W. C., Potter, S. S., Gualtieri, J. W., Roth, E. M. and Easter, J. R., Applied Cognitive Work Analysis: A Pragmatic Methodology for Designing Revolutionary Cognitive Affordances, Book Chapter, 2002.

[Endsley, 1995], Endsley, M. R., Toward a Theory of Situation Awareness in Dynamic Systems, Human Factors Journal, 37(1) , pages 32-64, March 1995.

[Endsley, Garland, 2000], Endsley, M. R. and Garland, D. J., Situation Awareness Analysis and Measurement, Lawrence Erlbaum Associates, Mahawah, New Jersey, USA, 2000.

[Giarratano, Riley, 1998], Giarratano, J. and Riley, G., Expert Systems - Principles and Programming, PWS Publishing Company, Boston, 1998.

[Ginsberg, 1993], Ginsberg, M., Essentials of Artificial Intelligence, Morgan Kaufmann, San Francisco, California, USA, 1993.

[Girard, 2004], Girard, LTC J., Defence Knowledge Management: A Passing Fad?, Canadian Military Journal, Summer 2004.

[Gómez-Pérez et al, 2004], Gómez-Pérez, A., Fernández-López, M. and Corcho, O., Ontological Engineering, Springer, London, 2004.

[Gruber, 1993], Gruber, T. R., A Translation Approach to Portable Ontology Specification, Knowledge Acquisition 5(2): 199-220, 1993.

[Jousselme, Maupin, Bossé, 2003], Jousselme,A.-L., Maupin, P. and Bossé, É., Uncertainty in a Situation Analysis Perspective, Proceedings of the 6th International Conference on Information Fusion, Cairns, Australia, 2003.

[Lambert, 2001], Lambert, D.A., An Exegesis of Data Fusion, Soft Computing in Measurement and Information Acquisition, in publication Edited L. Reznik and V. Kreinovich. Studies in Fuzziness and Soft Computing, Physica Verlag, 2001.

[Lambert, 2003], Lambert, D.A., Grand Challenges of Information Fusion, Proceedings of the 6th International Conference on Information Fusion. Cairns, Australia. pp. 213 – 219, 2003.

[Llinas, Antony, 1993], Llinas, J. and Antony, R. T., Blackboard Concepts for Data Fusion Applications, International Journal of Pattern Recognition and Artificial Intelligence, Vol 7, No. 2, 1993.

[Llinas et al, 2004], Llinas, J., Bowman, C., Rogova, G., Steinberg, A., Waltz, E. and White, F., Revisiting the JDL Data Fusion Model II, in Proceedings of the International Society on Information Fusion (ISIF) Conference, Fusion 2004, Stockholm, Sweden, June 2004.

[McIntyre, Gauvin, Waruszynski, 2003], McIntyre, S. G., Gauvin, M. and Waruszynski, B., Knowledge Management in the Military Context, Canadian Military Journal, Spring 2003.

[Merriam-Webster, 1981], Webster's Third New International Dictionary, Springfield MA: Merriam-Webster, 1981.

[Merriam-Webster, 2003], Merriam-Webster's 11th Collegiate Dictionary, Software, Version 3.0, 2003.

[Nonaka, Takeuchi, 1995], Nonaka, I. and Takeuchi, H., The Knowledge Creating Company, New York: Oxford, 1995.

[Nowak, Lambert, 2005], Nowak, C. and Lambert, D.A., The Semantic Challenge for Situation Assessments, Proceedings of the 8th International Conference on Information Fusion. Philadelphia, U.S.A., 2005.

[Polanyi, 1967], Polanyi, M., The Tacit Dimension, Garden City: Anchor, 1967.

[Powell et al, 2006], Powell, G.M., Matheus, C.J., Kokar, M.M. and Lorenz, D., Understanding the Role of Context in the Interpretation of Complex Battlespace Intelligence, Proceedings of the 9th International Conference on Information Fusion (Fusion 2006), Florence, Italy, 10-13 July 2006.

[Prusak, 2001], Prusak, L., “Where did knowledge management come from?”, IBM Systems Journal, 40(4), 2001. http://www.research.ibm.com/journal/sj/404/prusak.html

[Roy, 2001], Roy, J., From Data Fusion to Situation Analysis, Proceedings of the Fourth International Conference on Information Fusion (FUSION 2001), Montreal, Canada, August 7-10, 2001.

[Roy, 2007], Roy, J., Holistic Approach and Framework for the Building of Knowledge-Based Situation Analysis Support Systems, DRDC Valcartier Technical Report, TR 2005 - 420, 2007.

[Roy, Auger, 2007-A], Roy, J. and Auger, A., Knowledge Representation Concepts, Paradigms and Techniques for Use in Knowledge-Based Situation Analysis Support Systems, DRDC Valcartier Technical Report, TR 2006-755, 2007.

[Roy, Auger, 2007-B], Roy, J. and Auger, A., Reasoning Processes, Methods and Systems for Use in Knowledge-Based Situation Analysis Support Systems, DRDC Valcartier Technical Report, TR 2006-756, 2007.

[Roy, Auger, 2007-C], Roy, J. and Auger, A., Knowledge and Ontological Engineering Techniques for Use in Developing Knowledge-Based Situation Analysis Support Systems, DRDC Valcartier Technical Report, TR 2006-757, 2007.

[Russell, Norvig, 1995], Russell, S. and Norvig, P., Artificial Intelligence - A Modern Approach, Prentice Hall, New Jersey, 1995.

[Stefik, 1995], Stefik, M., Introduction to Knowledge Systems, Morgan Kaufmann Publishers, San Francisco, California, 1995.

[Steinberg, Bowman, White, 1998], Steinberg, A. N., Bowman, C. L. and White, F. E., Revision to the JDL data fusion model, Joint NATO/IRIS Conference, Quebec City, October 1998.

[Struder et al, 1998], Struder R., Benjamins, V. R. and Fensel, D., Knowledge Engineering: Principles and Methods, IEEE Transactions on Data and Knowledge Engineering 25(1-2): 161-197, 1998.

[Turban, Aronson, 1998], Turban, E. and Aronson, J. E., Decision Support Systems and Intelligent Systems, Fifth Edition, Prentice Hall, New Jersey, 1998.

[Waltz, 2003], Waltz, E., Knowledge Management in the Intelligence Enterprise, Artech House, Norwood, MA, 2003.

[Waltz, 2004], Waltz, E., Framing the “Hard” Threat - Implications for Data Fusion, Sensor Fusion Conference (Marcus Evans), Master Class: Novel Approaches to Hard Problems - Is Sensor Fusion Enough?, Crystal City, VA, 9 December 2004.

[White, 1987], White, F. E. Jr., Data Fusion Lexicon, Joint Directors of Laboratories, Technical Panel for C^3, Data Fusion Sub-Panel, Naval Ocean Systems Center, San Diego, 1987.

[White, 1988], Franklin E. White, Jr., A model for data fusion, Proc. 1st National Symposium on Sensor Fusion, vol. 2, 1988.

This page intentionally left blank.

List of symbols/abbreviations/acronyms/initialisms

AI	Artificial Intelligence
AKAMIA	Advanced Knowledge Acquisition for Maritime Information Awareness
ANS	Artificial Neural System
ARP	Applied Research Program
ASCAT	Analyse de la situation pour le commandant d'armée tactique
ASFI	Analyse de la situation et fusion d'information
ATR	Automatic Target Recognition
C2	Command and Control
C4ISR	Command and Control, Communication, Computer, Intelligence, Surveillance, and Reconnaissance
CBR	Case-Based Reasoning
CBSS	Computer-Based Support Systems
CCRP	Command and Control Research Program
CDS	Chief of the Defence Staff
CF	Canadian Forces
CKE-4-MDA	Collaborative Knowledge Exploitation for Maritime Domain Awareness
DM	Decision Making
DND	Department of National Defence
DoD	Department of Defense
DRDC	Defence R&D Canada
ECCESDM	Exploitation collaborative de la connaissance pour l'éveil situationnel du domaine maritime
ED	Expert du domaine
HTML	Hypertext Markup Language
HUMINT	Human Intelligence
ID	Identification
IKM	Information and Knowledge Management
INCOMMANDS	Innovative Naval Combat Management Decision Support
INFOSEC	Information Security
INTEL	Intelligence
IT	Information Technology
JCDS 21	Joint Command Decision Support for the 21st Century

JDL DIFG	Joint Directors of Laboratories' Data and Information Fusion Group
JIIFC	Joint Information and Intelligence Fusion Capability
KB	Knowledge Base
KID	Knowledge, Information, Data
KM	Knowledge Management
KR	Knowledge Representation
LF ISTAR	Land Force Intelligence, Surveillance, Target Acquisition and Reconnaissance
LHS	Left-Hand-Side
MDN	Ministère de la Défense Nationale
MSOC	Marine Security Operations Centres
OODA	Observe-Orient-Decide-Act
OPS	Operations
OPSEC	Operational Security
PRA	Programme de recherches appliquées
R&D	Research & Development
RHS	Right-Hand-Side
SA	Situation Analysis
SAIF	Situation Analysis and Information Fusion
SASS	Situation Analysis Support Systems
SATAC	Situation Analysis for the Tactical Army Commander
SAW	Situation Awareness
SME	Subject Matter Expert
S&T	Science and Technology
TDP	Technology Demonstration Program
TR	Technical Report
U.S.	United States

Distribution list

Document No.: DRDC Valcartier TR 2005-419

LIST PART 1: Internal Distribution by Centre:

1 - Director General
3 - Document Library
1 - Head/Information and Knowledge Management Section
1 - Head/Decision Support Systems Section
1 - Head/System of Systems Section
1 - Head/Optronic Surveillance Section
1 - LCol Pierre Lefebvre (DJCP CTG C4ISR Team Leader)
1 - Mr Claude Roy (SDA)
1 - Mr Jean Roy (author)
1 - Dr Alain Auger
1 - Mrs Régine Lecocq
1 - Mr Yaïves Ferland
1 - Dr Michel Gagnon
1 - Mr François Létourneau
1 - Mr Denis Gouin
1 - Mr Alain Bouchard
1 - Dr Luc Pigeon
1 - Mr Alain Bergeron
1 - Mr François Lemieux
1 - Mr Marc-André Morin
1 - Mr Alexandre Bergeron-Guyard
1 - Mr Réjean Lebrun
1 - LCdr Elizabeth Woodliffe
1 - Dr Pierre Valin
1 - Dr Anne-Laure Jousselme
1 - Dr Hengame Irandoust
1 - Dr Patrick Maupin
1 - Mr François Réhaume
1 - Mr Stéphane Paradis
1 - Dr Abderezak Benaskeur
1 - Dr Adel Guitouni
1 - Mrs Micheline Bélanger
1 - Dr Anne-Claire Boury-Brisset
1 - Mr Abder Sahi
1 - Mrs Catherine Daigle
1 - Mr Martin Blanchette
1 - Dr Richard Breton
1 - Maj Michel Gareau
1 - Mr Jean-Claude St-Jacques

1 - Dr Marielle Mokhtari
1 - Mr François Bernier
1 - Mr Michel Lizotte
1 - Mr Mario Couture
1 - Mr David Ouellet
1 - Mr Gaetan Thibault
1 - Mr Éric Dorion
1 - Mr Dany Dessureault
1 - Mr Jean Dumas
1 - Dr Alexandre Jouan
1 - Mr Benoît Ricard
1 - Mr Jean Maheux
1 - Mr Jim Cruickshank
1 - Mr Denis Dion

TOTAL LIST PART 55

LIST PART 2: External Distribution by DRDKIM

NDHQ
101 Colonel By Drive
Ottawa ON K1A 0K2

1 - Director – R&D Knowledge and Information Management (PDF file)
1 - Col Rick Williams (Associate DG RDP)
1 - Director – Directorate Science and Technology (Air)
Directorate Science and Technology (Land)
1 - Attn: Mr Craig Maskell
1 - Attn: Maj David Waller
Directorate Science and Technology (Maritime)
1 - Attn: Dr Pierre Lavoie (DSTM)
1 - Attn: Mr Greg Walker (DSTM 3)
1 - Attn: Mrs Laura Ozimek (DSTM 5)
Directorate Science and Technology (Policy)
1 - Attn: Mr Clément Laforce
1 - Attn: Mr Robert Webb
1 - Attn: Mrs Lisa Willner
1 - Director – Directorate Science and Technology (Human Performance)
Directorate Science and Technology (C4ISR)
1 - Attn: Dr Chris McMillan (DSTC4ISR)
1 - Attn: Mr Klaus Kollenberg (DSTC4ISR 3)
1 - Attn: Mrs Donna Wood (DSTC4ISR 4)
1 - Attn: Caroline Wilcox (DSTC4ISR 5)
Centre for Security Science (CSS)
1 - Attn: Dr Anthony Ashley – Director General
1 - Attn: Mrs Susan McIntyre – Knowledge Manager

1 - Attn: Director – CBRN Research and Technology Initiative (CRTI)
1 - Attn: Director – Public Security Technical Program (PSTP)
DRDC Centre for Operational Research and Analysis (CORA)
1 - Attn: Dr Kendall Wheaton
1 - Attn: Mr Dave Connell
1 - Attn: Mr Ron Funk
1 - Attn: Mr Paul Comeau
1 - Attn: Mr Paul Pulsifer (DGIP/DSTI)
1 - Cdr Stephen R. Peters – CMS – DGMFD – DMRS 2 (Maritime Security Section Head)
1 - LCdr Jim Day – CMS – DGMFD – DMRS 2 (Project Director MSOC)
1 - LCol John Langen – DPDOIS (Project Manager MSOC)
1 - LCol Jacques Hamel – DLCSPM 7 (PMO ISTAR)
1 - Capt(N) Darren Knight – CMP – CDA
1 - Mr Paul Morin – DCDS – CDI – D INT IM
1 - LCol Terrence Procyshyn – DJFC 4 (PD JIIFC
1 - LCdr Craig Dewar – DJFC (D/PD JIIFC)
1 - LCol Thomas Sullivan – ADM (IM) (PM JIIFC)
1 - Mr Robert McAulay – ADM (IM) (D/PM JIIFC)
1 - LCol Phil Jourdeuil – ADM (IM) – DGIMSD – DIMSP
1 - LCol Rick Johnston – VCDS – CFD – DJCP
VCDS – DGSC – DKM
1 - Attn: LCol Christopher Blodgett
1 - Attn: Mrs Karen Lahaise
1 - Attn: Mr Mike Crowell
1 - Cdr Haydn Edmundson – J3 Maritime – Support and Tasks – Canada Command HQ

DRDC Atlantic
9 Grove Street
Dartmouth NS B2Y 3Z7

1 - Dr Ross Graham – Director General – DRDC Atlantic (Scientific Advisor (Maritime))
1 - Mr Jim L. Kennedy
1 - Cdr Siegfried Richardson-Prager (SMO)
1 - Mr Jim S. Kennedy (Thrust Leader "11h")
1 - Dr Mark McIntyre
1 - Mr Dave Chapman
1 - Mrs Liesa Lapinski
1 - Mr William Campbell
1 - Mr Mark Hazen

DRDC Toronto
PO Box 2000
1133 Sheppard Avenue
Toronto ON M3M 3B9

1 - Dr Ross Pigeau
1 - Mrs Carol McCann (Thrust Leader "12o")
1 - Mrs Sharon McFadden

1 - Dr David Smith
1 - Dr Justin Holland
1 - Mr Kevin Trinh
1 - Dr. Peter Kwantes
1 - Ms Renée Chow

DRDC Ottawa
3701 Carling Avenue
Ottawa ON K1A 0Z4

1 - Dr Brian Eatock
1 - Mr Gary Geling
1 - Mr Rick Brown
1 - Mr Chris Helleur
1 - Mrs Barbara Waruszinsky
1 - Mr Jack Pagotto
1 - Mr Dan Brookes
1 - Mr David DiFilippo

Canadian Forces Experimentation Centre (CFEC)
3701 Carling Avenue
Ottawa ON K1A 0Z4

1 - Capt(N) Kevin Laing
1 - LCol Tom Gibbons
1 - LCol J.D. Graham
1 - LCdr Rob Elford
1 - Dr Phil Farrell
1 - LCol Tom Gibbons

Communications Security Establishment (CSE)
1500 Bronson Avenue
Ottawa, Ontario, K1G 3Z4

1 - Mrs Brigitte Hébert (Technology Forecasting)
1 - Mr Rodney Howes (R&D Partnerships)

Canadian Security Intelligence Service (CSIS)
P.O. Box 9732,
Postal Station "T",
Ottawa, Ontario,
K1G 4G4

1 - Mr King W. Jean

Canadian Police Research Centre
National Research Council
Building M-50, 1200 Chemin Montreal Road

Ottawa, Ontario
K1A 0R6

1 - Mr John Arnold (Chief Scientist)
1 - Mrs Claudette Moïse (Director, Library Services)

TOTAL LIST PART 77

TOTAL COPIES REQUIRED: 132

This page intentionally left blank.

DOCUMENT CONTROL DATA

(Security classification of title, body of abstract and indexing annotation must be entered when the overall document is classified)

1. ORIGINATOR (The name and address of the organization preparing the document. Organizations for whom the document was prepared, e.g. Centre sponsoring a contractor's report, or tasking agency, are entered in section 8.) Defence R&D Canada - Valcartier, 2459 Pie-XI Blvd. North, Quebec (Quebec), Canada, G3J 1X5	2. SECURITY CLASSIFICATION (Overall security classification of the document including special warning terms if applicable.) Unclassified

3. TITLE (The complete document title as indicated on the title page. Its classification should be indicated by the appropriate abbreviation (S, C, R or U) in parentheses after the title.)

A knowledge-centric view of situation analysis support systems (U)

4. AUTHORS (last name, followed by initials – ranks, titles, etc. not to be used)

Roy, J.

5. DATE OF PUBLICATION (Month and year of publication of document.) January 2007	6a. NO. OF PAGES (Total containing information, including Annexes, Appendices, etc.) 114	6b. NO. OF REFS (Total cited in document.) 45

7. DESCRIPTIVE NOTES (The category of the document, e.g. technical report, technical note or memorandum. If appropriate, enter the type of report, e.g. interim, progress, summary, annual or final. Give the inclusive dates when a specific reporting period is covered.)

Technical Report

8. SPONSORING ACTIVITY (The name of the department project office or laboratory sponsoring the research and development – include address.)

N/A

9a. PROJECT OR GRANT NO. (If appropriate, the applicable research and development project or grant number under which the document was written. Please specify whether project or grant.) Projects 11bh, 11hg, and 12of.	9b. CONTRACT NO. (If appropriate, the applicable number under which the document was written.) N/A
10a. ORIGINATOR'S DOCUMENT NUMBER (The official document number by which the document is identified by the originating activity. This number must be unique to this document.) DRDC Valcartier TR 2005-419	10b. OTHER DOCUMENT NO(s). (Any other numbers which may be assigned this document either by the originator or by the sponsor.) N/A

11. DOCUMENT AVAILABILITY (Any limitations on further dissemination of the document, other than those imposed by security classification.)

(X) Unlimited distribution
(   ) Defence departments and defence contractors; further distribution only as approved
(   ) Defence departments and Canadian defence contractors; further distribution only as approved
(   ) Government departments and agencies; further distribution only as approved
(   ) Defence departments; further distribution only as approved
(   ) Other (please specify):

12. DOCUMENT ANNOUNCEMENT (Any limitation to the bibliographic announcement of this document. This will normally correspond to the Document Availability (11). However, where further distribution (beyond the audience specified in (11) is possible, a wider announcement audience may be selected.))

Unlimited announcement

13. ABSTRACT (A brief and factual summary of the document. It may also appear elsewhere in the body of the document itself. It is highly desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall begin with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (S), (C), (R), or (U). It is not necessary to include here abstracts in both official languages unless the text is bilingual.)

Situation awareness has emerged as an important concept around dynamic human decision-making in both military and public security environments. Situation analysis is defined as the process that provides and maintains a state of situation awareness for the decision maker(s), and information fusion is a key enabler to meeting the demanding requirements of situation analysis in future command and control (C2) and intelligence support systems. This report presents a new perspective of situation analysis and information fusion (SAIF). The author proposes that developing and adopting a knowledge-centric view of situation analysis should provide a more holistic perspective of this process, leading to the development of better, more adequate SAIF support systems for the operational communities. Along this line of thought, the report provides a fair overview of many of the main issues around SAIF and knowledge-based systems in order to set up the foundational R&D framework for two projects that are just starting at Defence R&D Canada. Aspects being discussed include C2, intelligence, SAIF, support systems, knowledge, knowledge representation and management, reasoning processes, methods and systems, artificial intelligence, processing/computing/programming paradigms, and expert systems. The effort reported here constitutes some solid basis on which a long-term R&D program in SAIF should be built.

14. KEYWORDS, DESCRIPTORS or IDENTIFIERS (Technically meaningful terms or short phrases that characterize a document and could be helpful in cataloguing the document. They should be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location may also be included. If possible keywords should be selected from a published thesaurus, e.g. Thesaurus of Engineering and Scientific Terms (TEST) and that thesaurus identified. If it is not possible to select indexing terms which are Unclassified, the classification of each should be indicated as with the title.)

Situation Awareness, Situation Analysis, Information Fusion, Knowledge-Based Systems, Knowledge Representation, Expert Systems.

Defence R&D Canada

Canada's Leader in Defence
and National Security
Science and Technology

R & D pour la défense Canada

Chef de file au Canada en matière
de science et de technologie pour
la défense et la sécurité nationale

DEFENCE R&D DÉFENSE

WWW.drdc-rddc.gc.ca