Assessing Intelligent Information Systems for C2

Raymond A. Liuzzi, PhD
AFRL/IFTD
525 Brooks Road
Rome, New York 13441
Voice: 315-330-3577
Fax: 315-330-2563
Email: liuzzir@rl.af.mil

Joseph P. Cavano
Techbridge
176 Northland Circle
Rome, NY 13440

Abstract

This paper will present a vision for intelligent information processing for Command and Control (C2) in the 21st century This vision will describe: (1) the role, utilization and impact of knowledge base (KB) technology within an IIS, (2) the need to assess intelligent information system's (IIS), (3) a proposed methodology to assess IIS's, and (4) a model architecture environment, called the Integrated Knowledge Sharing Environment (IKSE) that is designed to integrate, test and evaluate IIS components/tools.

Introduction:

Today's decision makers and knowledge workers are caught in the middle between increasingly complex problem domains and a vast world in digital information spread across a multitude of databases (e.g., web pages, knowledge bases, XML documents, legacy databases and data warehouses). Physical access to the data is no longer dictated by geography (the Internet has changed that) but logical constraints still abound that hinder using available data in the decision making process.

Getting the *right* information to the decision-maker still remains a serious problem. Too often decision-makers are buried under a mass of data that current systems deliver with increasing frequency and quantity. Ironically, 'the right information' needed to solve the problem at hand is either lost in a morass of data or not delivered successfully because it is stored in a separate, perhaps incompatible, database. Consequently, even though information systems abound and data seems only a mouse-click away, neither humans nor computers can effectively handle the knowledge work that is demanded of them.

One explanation for this shortfall in decision-making capability is that we have too often only mechanized existing tasks and forms without re-engineering the entire process. By limiting computers into doing only the same things that we used to do manually, we have not realized their great potential and may have only complicated our lives.

The mechanization stage for information processing is characterized by single task perspective and the necessity of human control. Individual task orientation brings with it a “tunnel vision” that fails to bring the full power of a computer system to bear on the total problem. Since we use computer systems to handle only part of the greater mission (usually the portion that deals with well-structured problems and structured data); the human is then required to integrate each part of the problem, in whatever tasks remain. Humans are also required to decide which tools are needed for the task at hand and it is their responsibility to select and bring to bear the proper data to a set of disjointed tools and computer systems.

There are three challenges facing developers of IIS’s to improve decision-making and realize productivity gains. We need to (1) shift from a task orientation to a total mission orientation, changing the kinds of things we do because of computers. This requires (2) that we rethink our information processes, make them more intelligent, thereby relieving humans from integration. We must, (3) provide decision makers with the ability to access and use information and innovative intelligent information systems (IIS’s) tools/components that adapt and respond to changing environments and situations, and do so in a way that allows maximum mobility, adequate performance and necessary security.

If these challenges are met successfully, decision-makers and IIS’s will become part a unified process. As we achieve greater automation and better integration of information systems for how we do things, we will be embarking on a real information system revolution. To be successful, we must focus on a total mission perspective while making the information system more responsible for the underlying control.

Role of Knowledge Bases:

To understand the importance of knowledge bases in an IIS the following questions are addressed:

a. Why is a knowledge base important?
b. What is the role of a knowledge worker in an IIS?
c. What is the importance of human and computer interaction in terms of intelligent decision making?

Knowledge Explosion:

We find that a key IIS attribute, exploitation, enables a user to employ intelligent analysis, detect trends, and disseminate decision and information wherever it is needed. Today’s decision-maker is caught in the middle because neither humans nor computers can effectively handle sophisticated knowledge. Complex problem domains, large search spaces, lots of data but little information all contribute to the problem. Traditional data base technology cannot handle this problem because it addresses the management of data and not the management of the information process. This is why we must turn to emerging knowledge bases to effectively address the decision-making problems of tomorrow. Knowledge is the key in tomorrow’s C2 systems and at the core of this explosion is knowledge base technology.

Knowledge Base Technology:

There are three key components that comprise Knowledge Base Technology:

Knowledge acquisition addresses the ways in which a knowledge base is created from raw data and converted to information incorporating procedures and relationships through a mission perspective ontology. **Reasoning** implies ways that a knowledge base is used to infer results. **Learning** is the key to identifying how machine learning techniques can be used to generate new knowledge.

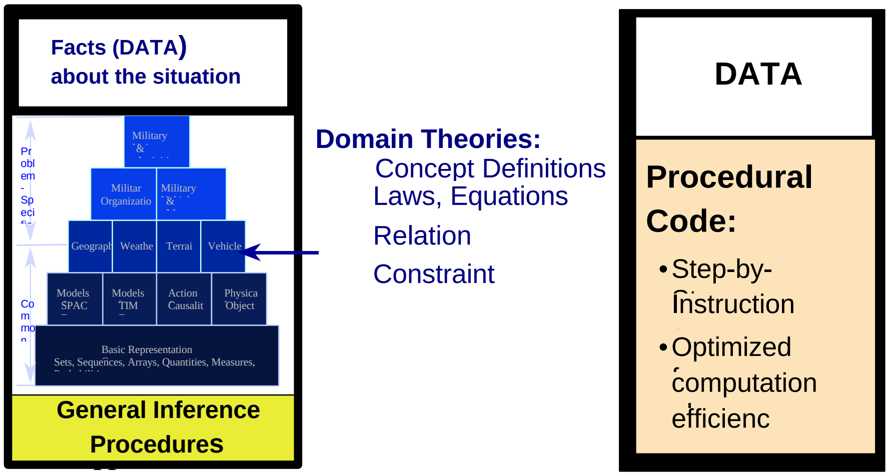
Data Bases Verses Knowledge Bases:

A knowledge base and a data base can be compared in a number of ways (Figure 1):

What is a Knowledge Base?

Definition: A set of representations of facts about the world. Each individual representation is called a rule or axiom. These rules/axioms are usually expressed in a knowledge representation language.

Knowledge Base | Data Base



Domain knowledge is explicitly stated in concept definitions, axioms and constraints

Domain knowledge is implicit in the programmer's step-by-step procedural instructions

Figure 1

1) One begins to develop a knowledge base via knowledge acquisition. A knowledge base consists of declarative knowledge (e.g. facts, ontologies, implication-rules). An upper ontology consists of the most general terms and concepts and their definitions, relationships and constraints. These include attributes like tangibility, being a set, etc. The core theories consist of general knowledge about the world including theories for space, time, causality, motion, belief, and objects described in a formal logical language. Knowledge bases handle data about a specific situation. Databases handle specific facts (e.g., the number of airplanes at a location).

2) A knowledge base is a compilation of information about the world, with the information broken into small independent pieces, which are each, represented in some machine-manipulable format (such as a logical axiom, an English sentence, or an if-then rule.) In a knowledge base, the domain knowledge is explicitly stated in concept definitions, axioms and constraints. In a data base, the domain knowledge is implicit in the programmer's step-by-step procedural instructions.

3) Subject Matter Experts (SME's) are needed to collect and organize knowledge in a knowledge base while clerks can handle data within a database.

4) Knowledge bases deal with high level abstractions (e.g., a statement that an F-111 is a type of airplane). Within this context, one can use a database and/or a knowledge base to retrieve information in much the same way. If we ask a question like "where is airplane number XYZ," both a database and knowledge base can directly look-up and retrieve the answer.

5) Knowledge bases can be used in a reasoning process to infer results directly. For instance a knowledge base can actively inform, diagnose, monitor, etc. If we ask a question like "Which air vehicles can move to a specific region?" a data base answer is found in a very different way from a knowledge base. For a database, a procedure must be written to determine the answer. However, a knowledge base can directly infer the answer from its set of facts and rules.

6) Knowledge base specialists focus on the knowledge level and database specialists focus on the data level.

7) Knowledge bases can be used in a learning process, which is much different from a database. For instance, a decision-maker uses a knowledge base and learns which allows more effective interaction as the process goes on. However, this learning area is still very much a critical research area and will become increasingly important to C2 systems as knowledge base usage increases.

Role of Knowledge Worker:

The knowledge worker is a key component in an intelligent information system.

"To make knowledge useful in a system, one cannot just "extract" the knowledge from the expert; one must structure it in such a way that it bears on the whole range of expected cases. Working with the expert, the knowledge worker must define the depth of representation, the expected limits of the system's explicit knowledge, the conditions under which the knowledge becomes inapplicable, etc. Knowledge is an artifact, worthy of design." (1)

Knowledge workers utilizing IIS's bring unique capabilities to the decision making process. These include:

1) Ability to provide information access in terms of the quantity and types of data available. This includes unstructured problems and data (including audio and video) and how best to integrate information from many sources.

2) Ability to see "the Big Picture." A knowledge worker must address process orientation and automate what needs to be done. A knowledge worker needs to understand the problems associated with a system sharing of knowledge instead of document sharing.

Humans verses Computers

Difficult technical challenges involve bridging the chasm between human logic and the formal logic of machines. Humans possess vast amounts of common sense acquired through years of experience, employ analogies, situational contexts, and reason using natural language and complex visualizations. Machines are constrained to techniques of mathematical logic, require precise representations and usually employ deductive theorem proving. The inherent complexity of managing and utilizing large knowledge bases grows exponentially as knowledge bases expand. Knowledge workers must accept and address this challenge in concert with their role with IIS's. It is in this area that (a) humans as knowledge workers and (b) computers with the ability to learn where a need to redefine roles in relation to 21st Century C2 systems must be examined.

Why Assess Intelligent Information Systems?

In order to help assess IIS's the following question must be addressed:

> **"What kinds of IIS and associated tools/components are needed to integrate data from multiple sources, transform this data into usable information, further formulate the information into knowledge, analyze and understand this knowledge, and then apply the knowledge for near real-time decision making?"**

Because of the rapid advancement of technology and the emergence of knowledge base capabilities, IIS's are appearing at a rapid rate. These systems all have a common theme in that knowledge is a critical component in accurate decision making. Some examples of IIS's that are just starting to emerge and are described below provide a crosscut view of IIS's and the types of capabilities they may be able to provide.

Examples of Intelligent Information Systems:

Cyc is a very large, multi-contextual knowledge base and inference engine (4,5,6,7). **Cyc** has a vast declarative represented array of knowledge about the world, to be added to, changed, manipulated, and used as the raw material or inputs for inferencing. **Cyc** uses logical deduction to combine assertions from its knowledge base to infer new assertions (which are then added into the knowledge base.) **Cyc** is being developed at Cycorp, under the guidance of President, Douglas Lenat.

Cyc is currently being used in the DARPA High Performance Knowledge Base (HPKB) project (8). HPKB is concerned with developing innovative technologies supporting the construction of knowledge bases, ontology's, and associated libraries of problem-solving strategies. A DARPA follow-on program, Rapid Knowledge Formation (RKF) (18), is addressing how domain experts can rapidly build up knowledge bases using their expert knowledge themselves, without help from knowledge engineers, by using dialogue, analogy, and similar methods. RKF plans to use **Cyc** for new ways of developing knowledge. Commercially, **Cyc** is being used by Lycos and Hotbot for smart web searching. **Cyc** is also being explored as a possible power source for speech understanding, data base integration, and knowledge management applications.

(2) Enkia Corporation's **IRIA** (Information Research Intelligent Assistant) technology (9,10,11,12,13) is a revolutionary new approach to intelligent information management. IRIA is a proprietary artificial intelligence (AI) toolkit that provides intelligent information management (IIM) capabilities for interactive knowledge-intensive applications, enabling developers to use recent techniques from artificial intelligence and cognitive science to provide advanced information capabilities to users in a natural, user-friendly manner. An IRIA-based web research assistant can unobtrusively monitor a user's browsing activity and suggest "hot" items, enabling users to quickly drill through hundreds of results to find the ones relevant to their interests. IRIA is based on a novel information retrieval architecture that uses memory models from cognitive science for search and mapping, embedded in a knowledge discovery and presentation engine using techniques from artificial intelligence and machine learning.

(3) **CoBase** is a unique product that can provide approximate answers if there are no exact answers. Querying conventional databases requires users to have detailed knowledge, such as schemas and attributes about the databases. If no exact answers are found, it returns no answer. **CoBase** allow users to pose queries with conceptual terms, which mimic human cognitive process. **CoBase** was developd at UCLA and Relaxation Technologies under the guidance of Dr. Wesley Chu. (2,3) **CoBase** utilizes a user and context sensitive knowledge base called Type Abstraction Hierarchies (TAH), which can be generated automatically from the data sources. Therefore, it is scalable to large systems. **CoBase** supports ODBC and JDBC interfaces and can be run on top of most commercial relational database systems (e.g. Oracle, Sybase, Teradata, etc.) and can also interface with commercial map servers (e.g. ESRI map objects). CoBase has been used (1) for electronic warfare for approximate matching of emitter signals, (2) in logistic planning applications for query relaxation (e.g., similar to) and spatial relaxation (e.g., near to) on the map server, (3) for integration of heterogeneous databases for approximate matching of attributes and views, (4) in the transportation domain for query relaxation and association, and (5) for retrieving medial images by approximate matching image (X-ray, MRI) features and content.

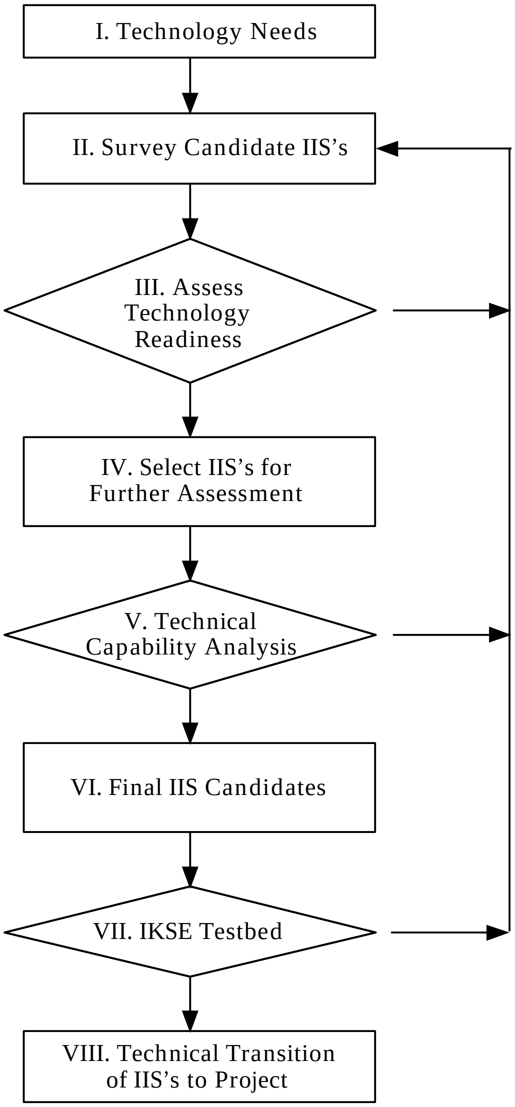
Methodology to Assess an Intelligent Information System

A methodology to assess IIS's is depicted in Figure 2. This methodology is based on surveying the emerging worlds of IIS tools/components and injecting evaluation criteria at several points to match an IIS tool/component with a specific technical need.

Selecting the right technology to use in building any complex system is arguably the most important decision that anyone will make that system's development. Examples of catastrophic failures in software endeavors abound. The Denver airport "opened a couple of years late because the hordes of C programmers couldn't make the computerized baggage handling system work (it was eventually scrapped)….[while] air traffic controllers up in the tower are using software from the 1960s because the FAA can't get their new pile of C code to work" [16].

Future Command and Control systems will combine many diverse software technologies, especially in the Intelligent Information System components, and the failure of any one of them could lead to disastrous results for the entire system. These systems must be built with quality if they are intended to perform their intended functions successfully. Without a common basis for evaluating the technologies that will be incorporated, management decision-making will be too haphazard to adequately plan and implement a large Command and Control system.

As a result, we recommend the following eight-step assessment methodology for selecting and incorporating intelligent information system technology for C2 systems. This methodology includes three different ways for evaluating specific technology to determine whether it should be transitioned into a project. The authors employed an earlier version of this methodology to assess emerging intelligent information systems for an Air Force partner in 1999, examining over thirty technologies and ultimately recommending five for further transition [14]. The methodology presented here has been slightly revised based on lessons learned from that experience.


I. Technology Needs
II. Survey Candidate IIS's
III. Assess Technology Readiness
IV. Select IIS's for Further Assessment
V. Technical Capability Analysis
VI. Final IIS Candidates
VII. IKSE Testbed
VIII. Technical Transition of IIS's to Project

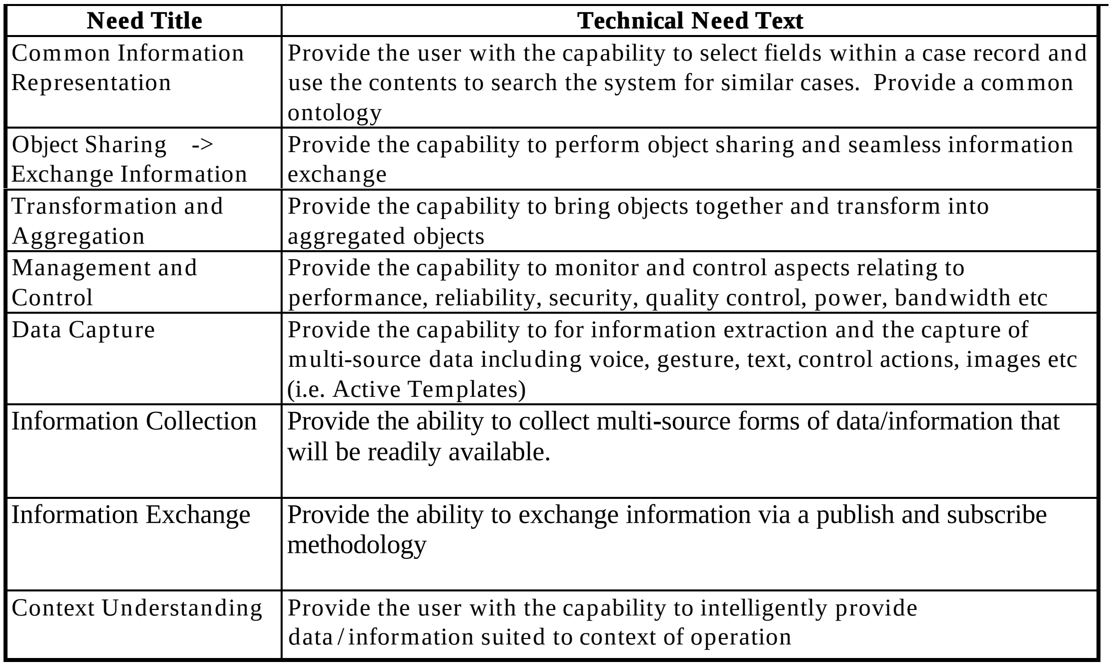
Eight-pass Assessment Methodology for Intelligent Information Systems
Figure 2

I. Technology Needs: To begin using this methodology, a project must identify its technology needs. A key aspect in this step is viewing the needs of the project over its lifetime. An easy trap to fall into at the beginning is to take a too limited view of technology needs. If the needs are stated too conservatively, it is possible the next project will only duplicate what has already been accomplished and not take advantage of new technology capabilities. The output of the technology needs step will identify the characteristics or factors that will mark the success of the overall system. These "Critical Success Factors," based on the requirements of the ultimate domain application, become the basis for selecting Intelligent Information System components. How well a new technology can satisfy the needs of a project's Critical Success Factors determines the degree to which that technology should be incorporated. For example, the success factors for an intelligent information system might require it to be interoperable, capable of dealing with extremely large databases, and need a high degree of usability.

In order to illustrate how this methodology can be employed we have chosen the concept of a Joint Battlespace Infosphere (JBI) (15) as an example.   The JBI concept is envisioned as a combat information management system that provides individual users with the specific information required for their functional responsibilities during crisis or conflict. The JBI concept incorporates the following critical success factors:

1. Information exchange
2. Transforming data to knowledge
3. Distributed collaboration
4. Assigned unit incorporation

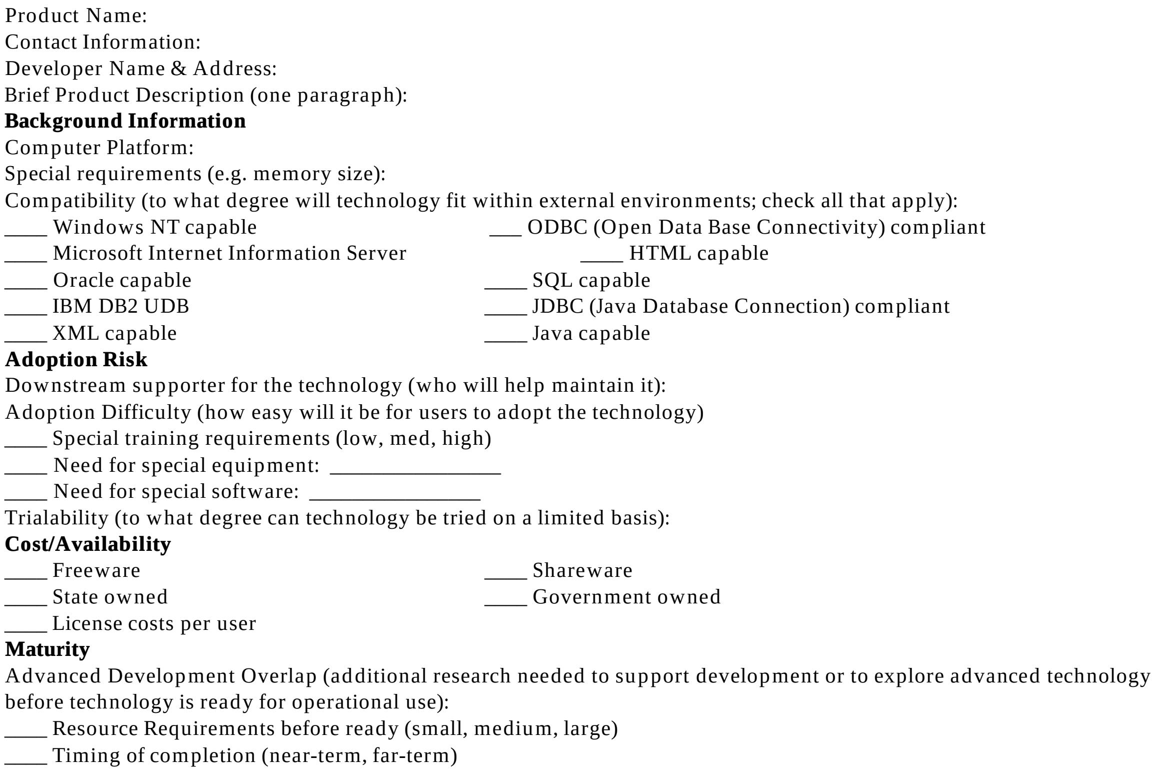
These factors translate into many different kinds of technical needs. We have identified several technical needs that may be related to an IIS capability and are illustrated Table 1.



Need Title	Technical Need Text
Common Information Representation	Provide the user with the capability to select fields within a case record and use the contents to search the system for similar cases. Provide a common ontology
Object Sharing -> Exchange Information	Provide the capability to perform object sharing and seamless information exchange
Transformation and Aggregation	Provide the capability to bring objects together and transform into aggregated objects
Management and Control	Provide the capability to monitor and control aspects relating to performance, reliability, security, quality control, power, bandwidth etc
Data Capture	Provide the capability to for information extraction and the capture of multi-source data including voice, gesture, text, control actions, images etc (i.e. Active Templates)
Information Collection	Provide the ability to collect multi-source forms of data/information that will be readily available.
Information Exchange	Provide the ability to exchange information via a publish and subscribe methodology
Context Understanding	Provide the user with the capability to intelligently provide data/information suited to context of operation

Table 1 Potential Technical Needs)

II. Survey Candidate IIS's: The next step is to develop a list of candidate intelligent information systems (IIS). Every organization should be keeping abreast of technology advances, whether in-house researchers or external providers drive them. This on-going technology awareness should provide a ready list of emerging technologies that can be matched against the technology needs. If a technology can provide a competitive advantage to the technical needs of the system, it should be included at this step. A survey of each technology should be performed that includes specific needs and these needs are analyzed against what the technology can offer. Criteria included in the survey seek to assess maturity of the IIS technology and the extent to which the IIS can meet an advanced development scenario. A sample of a possible survey form (without technology needs) is illustrated in Figure 3.



Product Name:
Contact Information:
Developer Name & Address:
Brief Product Description (one paragraph):

Background Information

Computer Platform:
Special requirements (e.g. memory size):
Compatibility (to what degree will technology fit within external environments; check all that apply):

____ Windows NT capable
____ Microsoft Internet Information Server
____ Oracle capable
____ IBM DB2 UDB
____ XML capable
___ ODBC (Open Data Base Connectivity) compliant
____ HTML capable
____ SQL capable
____ JDBC (Java Database Connection) compliant
____ Java capable

Adoption Risk

Downstream supporter for the technology (who will help maintain it):
Adoption Difficulty (how easy will it be for users to adopt the technology)
____ Special training requirements (low, med, high)
____ Need for special equipment: ________________
____ Need for special software: ________________
Trialability (to what degree can technology be tried on a limited basis):

Cost/Availability

____ Freeware
____ State owned
____ License costs per user
____ Shareware
____ Government owned

Maturity

Advanced Development Overlap (additional research needed to support development or to explore advanced technology before technology is ready for operational use):
____ Resource Requirements before ready (small, medium, large)
____ Timing of completion (near-term, far-term)

Figure 3: Candidate Collection Form

III. Assessing Technology Readiness: The third step is a broad cut for determining whether a technology is worthy of further consideration by determining its Technology Readiness Level (TRL). The TRL is a judgement of the maturity level to which the system has or will reach. These levels begin with basic research concepts and go as far as its actual success for a specific mission. To adequately determine the maturity of the technology a technology readiness level model adapted from a GAO report (18) is then invoked (see Table 2). This model describes nine different levels of maturity, ranging from the lowest level of technology readiness when scientific research begins to be translated into applied research and development to the highest level where an actual system using the technology was "flight proven" through successful mission operation. The GAO report suggested that the commercial world usually accepted a maturity level of seven (i.e., system prototype demonstration in an operational environment) when making a decision about incorporating a new technology. However, those decisions were based on the understanding that additional technology development would be conducted.

Technology Readiness Level	Description
1. Basic principles observed and reported.	Lowest level of technology readiness. Scientific research begins to be translated into applied research and development. Examples might include paper studies of a technology's basic properties.
2. Technology concept and / or application formulated.	Invention begins. Once basic principles are observed, practical applications can be invented. The application is speculative and there is no proof or detailed analysis to support the assumption. Examples are still limited to paper studies.
3. Analytical and experimental critical function and / or characteristic proof of concept.	Active research and development is initiated. This includes analytical studies and laboratory studies to physically validate analytical predictions of separate elements of the technology. Examples include components that are not yet integrated or representative.
4. Component and / or breadboard validation in laboratory environment.	Basic technological components are integrated to establish that the pieces will work together. This is relatively "low fidelity" compared to the eventual system. Examples include integration of "ad hoc" hardware in a laboratory [and alpha software prototype, perhaps produced by university].
5. Component and / or breadboard validation in relevant environment.	Fidelity of breadboard technology increases significantly. The basic technological components are integrated with reasonably realistic supporting elements so that the technology can be tested in a simulated environment. Examples include "high fidelity" laboratory integration of components.
6. System / subsystem model or prototype demonstration in a relevant environment.	Representative model or prototype system, which is well beyond the breadboard tested for TRL 5, is tested in a relevant environment. Represents a major step up in a technology's demonstrated readiness. Examples include testing a prototype in a high fidelity laboratory environment or in simulated operational environment. [Beta software version]
7. System prototype demonstration in an operational environment.	Prototype near or at planned operational system. Represents a major step up from TRL 6, requiring the demonstration of an actual system prototype in an operational environment, such as in an aircraft, vehicle or space. Examples include testing the prototype in a test bed aircraft. [Final beta software tested in operational environment].
8. Actual system completed and "flight qualified" through test and demonstration.	Technology has been proven to work in its final form and under expected conditions. In almost all cases, this TRL represents the end of true system development. Examples include developmental test and evaluation of the system in its intended weapon system to determine if it meets design specifications.
9. Actual system "flight proven" through successful mission operations.	Actual application of the technology in its final form and under mission conditions, such as those encountered in operational test and evaluation. In almost all cases, this is the end of the last "bug fixing" aspects of true system development. Examples include using the system under operational mission conditions.

Table 2 Technology Readiness Levels and Their Definitions

Once everything is examined, a readiness level can be chosen to determine appropriate fitness of a system. Only technology that has successfully performed at a specified high level, depending on the goals and risk tolerance of the project being targeted, should be considered for the next evaluation phase.

IV. Select IIS's for Further Assessment: The output of the third step would be subset list of candidate IIS's. In addition to those technologies that passed the 'rigid' Technology Readiness Level, truly promising technologies that could make significant contributions when they are ready should also be on this list. Major information systems, like Command and Control systems, are not built and forgotten but evolve over time. This step allows technologies to be included as the system evolves even though they made not be ready for the first implementation.

V. Technical Capability Analysis: Next, a second evaluation process is performed that more concretely compares technical capability with technical need. The orientation shifts from 'what may be possible' to a more specific type of examination. The Technical Capability Analysis uses results of the IIS survey of perspective technologies to help specify how the new technology will handle these needs. Criteria included in the survey seek to assess the environment within which the IIS operates, status of intellectual-property rights, unique features/drawbacks, and effort required to transition. Also, conversations with the technology providers is a key part of this step which begin the process of informing technology providers what is needed and obtaining their inputs for how their technology might play a role in meeting the needs of the project. This step, as envisioned, is ad hoc but the authors are seeking ways quality evaluation metrics may be used to make it more rigorous.

VI. Final IIS Candidates: After this process is complete, a final candidate list of IIS's can be made. As before, this list must still be broad enough to include technologies for further consideration even though they may not be ready for immediate implementation.

VII. Integrated Knowledge Sharing Environment: Till this point, the assessment has all been based on survey results, interviews, experience and judgement. However, before final decisions about whether to include a technology, first hand analysis is required. We propose an Integrated Knowledge Sharing Environment (IKSE) Test-bed (Figure 5) for incorporating new technology and testing it in an actual setting that is similar to the ultimate domain that is being targeted. The IKSE can be used for component/tool testing of various IIS technology as an integrated process to access information, devise alternatives, select best choices, and communicate results.

The IKSE serve's as a key interface between the decision maker and complex C2 missions and large, complicated knowledge bases. It will use domain knowledge of the C2 application to gain access to the necessary information space and to interface to other components that correspond to the decision making process and actively help in problem solving and verifying decisions by abstracting information from heterogeneous data sources, including knowledge bases, semi-structured databases, and unstructured text and image files. By achieving a transparent integration of information access and intelligent decision making processes within a mission-perspective, the environment will help a decision maker become more effective in selecting information, making and evaluating decisions and coordinating those decision with diverse team members using key IIS components/tools.

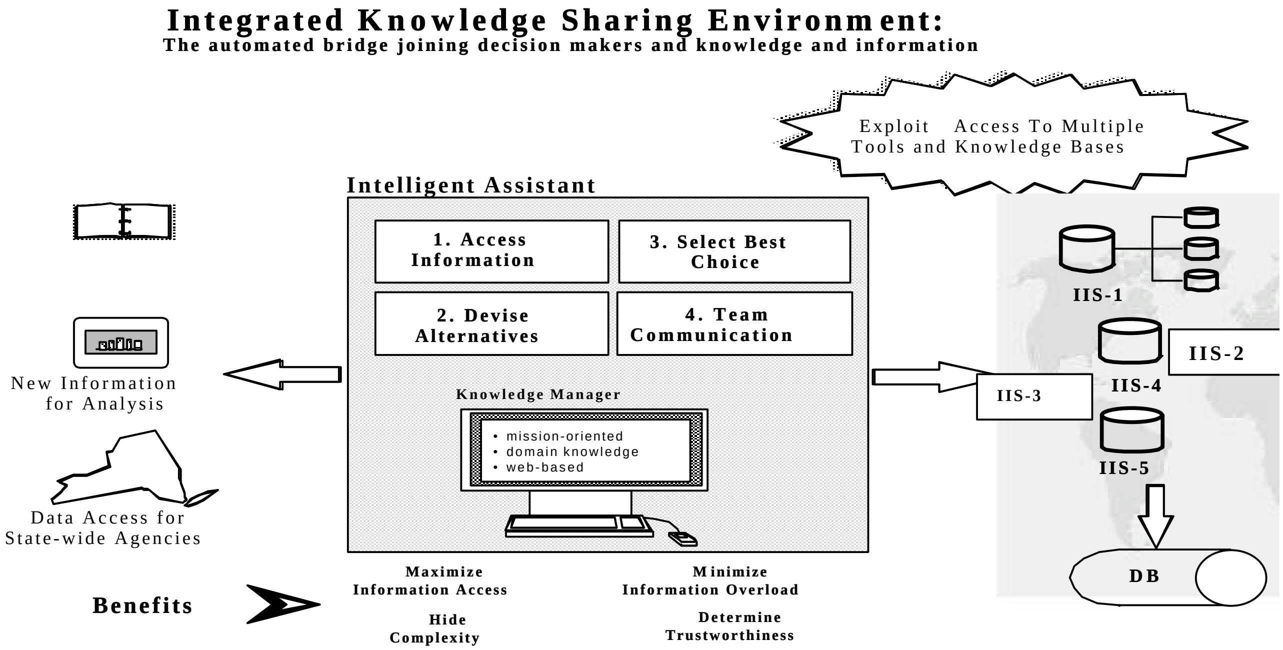

Integrated Knowledge Sharing Environment:
The automated bridge joining decision makers and knowledge and information
Exploit Access To Multiple Tools and Knowledge Bases
Intelligent Assistant
1. Access Information
3. Select Best Choice
2. Devise Alternatives
4. Team Communication
Knowledge Manager
• mission-oriented
• domain knowledge
• web-based
New Information for Analysis
Data Access for State-wide Agencies
IIS-1
IIS-2
IIS-3
IIS-4
IIS-5
DB
Benefits
Maximize Information Access
Minimize Information Overload
Hide Complexity
Determine Trustworthiness

Figure 5 Integrated Knowledge Sharing Environment

The IKSE is designed to provide a paradigm shift away from basing decisions on stand-alone documents and isolated information systems to a collaborative decision making process, accessible through internets and intranets. Web-accessible IIS components defined for this environment would aid in shifting from task-by-task orientation and mechanized forms to a mission orientation by automating and improving the decision making process. The ultimate goal of the environment is to help decision makers avoid information overload while quickly and easily navigating through information, regardless of location, and using that information more effectively to make better decisions. The payoff of the IKSE will be an improved capability for reducing information integration complexity. In addition, meeting the performance requirements for timely access to command and control information will lead to development of better systems.

Within the IKSE, decision-makers could have at their disposal a host of knowledge discovery, data mining and statistical analysis tools/components to help verify solutions and evaluate and recommend final choices. For the JBI example, an ontology describing the problem space could be included within the IKSE. Also for the JBI example a specified knowledge representation and structure and features to mimic large scale object type storage that would enable test, prediction and evaluation of transaction rates that would be needed for a typical scenario.

The payoff of the IKSE will be an improved capability for reducing information integration complexity and promoting interaction among IIS's. Not only will the IKSE promote additional understanding of the technology through actual use, it will also be playing in concert with other technologies integrated within in the testbed. Since, ultimately, all these technologies must be able to be integrated together operationally, the testbed provides a way to reduce potential risk.

VIII. Technical Transition: Our motto is**, "If you don't plan for technology transition, it probably won't happen!**" This is a result of both author's experience over the years with trying to move technical innovation to C2 application domains. The technologies being considered for the JBI example provide a paradigm shift from traditional information management policies. Will users be ready for this shift? This must be properly addressed in any transition. We must recognize that the best technology is not what always wins in the field. For those technologies that are recommended for incorporation into a project, a technical transition plan should be devised to insure that the technology could be successfully applied. Part of the technical transition will be keeping an eye on all the promising technologies that were examined but were left off the list, not because of technical deficiencies but rather because of maturity and readiness. Over the life of the project, these technologies should be groomed if they are judged worthy of making a competitive difference in the future.

A simple transition plan is proposed which employs a technology broker to facilitate the technology transfer. For this plan one must accomplish three functions (1) identify mission needs and barriers;   (2) establish viable pilot tests and identify necessary infrastructure and (3) provide training and on-site support capability and leverage success in new development opportunities. Within this context, there should be direct communication between the technology consumer and technology producer. A technology broker should act as the integrator between the consumer and producer. In this role the broker directs proposals and needs to the producer. These needs are fostered from the consumer. Solutions and proposals are directed to the producer. Feedback is provided and eventually solutions are created. The broker also identifies issues and resolves any barriers that may result. The broker oversees a maturation of the process and eventually assists in the implementation and installation.

Summary:

Using the results of the candidate technology survey and the use of the TRL table, candidate IIS's can be identified for potential capabilities meeting defined technical needs. The IIS's that satisfy a predetermined level are then passed to the next evaluation stage where a technical capability analysis is performed. Additional survey results and actual interfaces to the IIS developers are critical in this phase. In this phase key decisions are made to generate a final set of IIS's to be considered. This set of IIS are then evaluated in an IKSE laboratory environment. Once this evaluation is complete and key IIS's are chosen, technology transition needs must be addressed.

If the proposed methodology is followed and the transition is appropriately made overall risk will be reduced and C2 systems will be the beneficiary of new and possible revolutionary technology advances.

Final Recommendations:

(1) Explore new and emerging IIS technologies that can have the greatest long-term impact.
(2) Use the proposed methodology to evaluate emerging IIS tools/components. Establish a knowledge acquisition process well in advance of intended implementation.

(3) Explore development of the Integrated Knowledge Sharing Environment (IKSE). Create focused C2 scenarios to assess and evaluate capabilities.
(4) Establish a "technology broker" to facilitate communications between technology producer and technology consumer of new IIS technology.

References

1. Daniel Bobrow et al, "Expert Systems: Perils and Promise," Communications of the ACM, September 1986, Vol. 29, No.9

2. Wesley W. Chu, Scalable and Extensible Cooperative Information Systems, AFRL-IF-RS-TR-1999-166, August 1999

3. CoBase Home Page @ Relaxation Technology (http://www.rlxtech.com)

4. Cycorp's home page (http://www.cyc.com)

5. *Knowledge Source Documentation* by Eva Pierce

6. *Artificial Intelligence* (third Addition) by Patrick Henry Winston

7. *Building Large Knowledge-based Systems* by Douglas.B.Lenat

8. P. Cohen, R. Schrag, E. Jones, A. Pease, A. Lin, B. Starr, D. Gunning, M. Burke, "The DARPA High-Performance Knowledge Bases Project", American Association for Artificial Intelligence, Winter 1998

9. Green, H. (1999) The Information Gold Mine. *Business Week e.biz*, July 26 1999, pp. EB17-30

10. Internet World (1998). Getting there, or not: Why search is so ineffective. *Internet World Online Edition*, February 23, 1998.

11. Lawrence, S. & Giles, C.L. (1999). Accessibility of information on the web.
McWilliams, B. 1998. Search engine to sell top positions on result lists. *PC World Online*, February 23, 1998. *http://www2.pcworld.com/news/daily/0298/9802233173204.html*

12. Pitta, J. (1999). Forbes, August 23, 1999, pp76-77.

13. Ram, A., & Francis, A. G. 1996. Multi-Plan Retrieval and Adaptation in an Experience-Based Agent, In D. B. Leake, editor, *Case-Based Reasoning: Experiences, Lessons, and Future Directions*, AAAI Press.

14. R. A. Liuzzi, J. P. Cavano, "Assessing Intelligent Information Systems for CASE Management and Intelligence System (CMIS) for New York State", Report for NYSTEC, Dec 1999

15. United States Air Force Scientific Advisory Board, “Report on Building the Joint Battlespace Infosphere”, SAB-TR-99-02, Volume 1: Summary, Dec 1999

16) Philip and Alex's Guide to Web Publishing, Greenspun, Philip, Morgan Kautman Publishers, 1999.

17) M. Burke, DARPA Rapid Knowledge Formation BAA 99-15, Commerce Business Daily, 3 September 1999

18) "Best Practices," GAO Report, GAO/NSIAD-99-162, 1999