Auditable Intelligence Wins
(Policies for Future Systems)

I. Introduction:

It is a competitive world. To compete, one is faced with a continuous flow of issues that need to be addressed. Opposition forces (commercial, military, and political) compete with new tactics and new products, each creating their own challenges. There are constant pressures to do more with less, and at the same time perform better and faster. Even with advanced training techniques the capacity of individual humans to keep up with the new and more complex challenges is limited. Competitive threats from countries with an increased educated workforce cannot be met with manpower alone. There needs to be a way to capture and amplify human reasoning skills so that the United States can compete and succeed. The alternatives to success are not acceptable.

To move from hypothetical to real solutions, the approach must be explainable, auditable, flexible, extensible, and correctible. Solutions to critical (ex: life and death) problems demand that they be 100% explainable and auditable.

In some cases the responsibilities that have historically been dependent on skilled humans will have to be mechanized (automated). The Navy's Sea Power 21 vision, the Air Force Vision 2025, and the Army Future Combat Systems highlight the expectations of the future:

- do more with less
- keep humans out of harm's way
- mass produce human-like capability
- avoid human-error
- provide faster response
- utilize new information sources
- handle new problem sets

Solutions cannot be evaluated from capability alone. Life cycle cost must be considered and balanced against capability.

In the past, the ability to exercise judgment has separated humans from computerized systems. Knowledge Enhanced Electronic Logic (KEEL) is a platform technology that allows human-like reasoning to be captured, tested, packaged, audited and explained. It can be deployed in software systems or embedded in small real-time devices. It provides a new way to process information that emulates human judgment and reasoning. KEEL is supported with a "dynamic graphical language" that allows domain experts (not mathematicians) to define solutions to dynamic, non-linear, inter-related, multi-dimensional problem sets with relative ease. Humans (domain experts) create the cognitive models, so all decisions and actions made by systems and devices with embedded KEEL cognitive engines can be traced back to humans.

II. Alternative Information Processing Models:

Rule-Based Systems

Ever since digital computers have been created (the von Neumann architecture), computers have been processing rules. Numerous languages and packaging techniques have been developed to process IF | THEN | ELSE sequential logic. Functional decomposition and object oriented programming focused on code reuse and modularity. High level languages streamlined the process. Rule-based systems are fundamental to most automated solutions available today. Many initial attempts at artificial intelligence used forward and reverse chaining approaches to integrate textual rules and qualitative measurements.

As these systems were challenged in order to address more complex and dynamic systems, the cost to develop and maintain them escalated. Complex mathematical formulas have been and are required to address non-linear systems. There are extremely high costs associated with translating ideas into mathematical formulas and then into software "code" before the ideas can be tested. The conventional approach has been to continually develop faster and faster computers with more and more memory in order to keep up with the more complex rule sets (programs). In the past, once complex solutions were developed, the domain expert with the idea (or the person responsible for the policy that was to be captured) had almost no chance of understanding *how* their ideas were actually implemented. Extending complex, inter-related, rule-based systems has the potential of introducing "bugs" into the system that (in safety critical systems) have the potential of catastrophic errors.

There will be those that look backward and are satisfied with what we can do with the tools and techniques that have created the complex systems that exist today. There will be others that look forward in search of new technologies and tools needed to satisfy the ever more complex problems of the future.

Probability-Based Systems

An extension of rule-based systems is to include probability within the rules. This allows the rules to process the IF | THEN | ELSE logic within ranges. This is a common practice for many medical systems.

An issue with probability based systems is that it is difficult to obtain good statistics from non-linear systems. For example, regarding a selected population, the designer is often forced to segment the population into boxes that can be measured independently. For example, in the medical space regarding age groups, one may find that one day you are in one box and the next day you are in a completely different box.

In general, statistics and probabilities are good but they cannot be trusted unless the designer has a clear understanding what they mean and how accurate they are in reference to the problem.

Pattern Matching Systems (and Learning Systems) (Biological based models)

Artificial Neural Net (ANN) based systems are modeled after the structure of the human brain. ANN learns by being trained on known patterns. The addition of genetic techniques allows these systems to learn based on trial and error. This is the "holy grail" of some academics who want to create human-like devices that can learn on their own and solve problems in new ways. They feel that this may lead to new ways to solve problems. Pattern matching systems can also be used when humans do not really understand the interactions between inter-related variables. These systems just interpolate between taught, or learned, sets of inputs and outputs.

Autonomous learning systems have the potential of learning "bad" things. Because they are just interpolating between patterns, there is no real auditability. They cannot explain what they are doing. There is no "morality" in ANN based systems. There is no traceability. If an ANN-based system makes an error, it cannot be traced to anyone responsible, unless one points to the organization doing the training (which might suggest that the organization did not do sufficient training).

Training of ANN-based systems is a costly exercise. It can be <u>very</u> costly if the system is complex. In addition, there is still the potential of missing the critical pattern. Adding a new variable to an ANN-based system will likely require the entire system to be retrained on the full set of patterns.

Humans employ pattern matching as they uniquely observe and learn through their life experiences. Because humans are not mass produced with duplicate experiences (duplicate memories), there is limited risk. When one human exhibits bad behavior, that human can be isolated to limit the impact to the global population. If one application of this technology (ANN) is to mass produce a solution, the impact of mass producing a solution with critical side affects that are untraceable should not be acceptable.

Fuzzy Logic Based Systems

Fuzzy logic is based on "linguistic uncertainty". It provides a way to establish "values" from terms used by humans as part of their "language" such that those values can be used to drive computerized systems.

While the results of fuzzy logic systems are repeatable, their designs are not necessarily traceable (from a comprehension standpoint). The "fuzzification" process translates information through a "geometric" participation scheme that is as much of an art as it is a science to develop. The designer subjectively "chooses" a geometric pattern.

Fuzzy logic based systems are seldom used in safety critical systems because they cannot effectively be explained in a court of law.

Fuzzy logic designs are seldom system-wide solutions as the terminology used to describe / model one part of the system is likely to be different from another. This means that each element might have different modeling approaches.

III. Knowledge Enhanced Electronic Logic

KEEL Technology includes a tool that embodies a "dynamic graphical language". It allows one to define functional relationships that equate to "judgment and reasoning".

A user (domain expert) interacts with the tool. The user stimulates inputs and observes how the entire system responds. This interactive development process allows the user to see the system *think* (interpret information and balance alternatives). When the system performs the way the user wants it to perform, the user can package the cognitive model as a KEEL cognitive "engine".

The KEEL language is truly "dynamic". The ability to "see" the system process information is radically different from a textual language that is processed sequentially. One can see how all factors of the problem sets are valued under any set of input conditions.

Unlike rule-based systems where a programmer defines rules to get answers, with the KEEL language, a domain expert defines how factors are inter-related and observes the answers. Developing KEEL-based systems is an iterative process as one continually adds new factors into the decision-making model and tests how the factors interact with other factors. The systems evolve; getting more and more capable over time.

KEEL Technology targets applications that are expected to evolve as their operating environment is expected to change. Humans are in control all the time. Every decision and action is 100% auditable.

KEEL Technology is supported with formal user's documentation, training material, code walkthrough documentation, system engineering tools, and auditing tools for reverse engineering decisions and actions made by devices and autonomous systems.

IV. Conclusion:

Conventional rule-based systems, probability-based systems, biological-based systems, and fuzzy logic based models do not satisfy all of the necessary requirements in order to address the complex, dynamic problem sets that exist now and will exist in the future. KEEL Technology was specifically created to address judgmental types of problems; where sometimes conflicting alternatives need to be balanced (risk/reward, cost/schedule, tactics/strategy, one life against others, etc.).

The KEEL dynamic graphical language allows the rapid development of conceptual models that can be tested to address complex, dynamic situations. Solutions to complex problem sets require the non-linear application of selective resources. These solutions can be developed and tested by domain experts / policy makers, without requiring the services of skilled mathematicians. This allows the domain expert to focus on solving / modeling solutions to problems rather than attempting to explain a solution to a mathematician who must then explain the solution to a software engineer to embed in simulations or devices.

KEEL "tools" can be used to package the cognitive models as KEEL "Engines" that can be installed in almost any architecture (from small, real-time stand-alone devices like autonomous weapon systems, to large, distributed grid-based systems). KEEL "engines" equate to an analog computer and to a complex formula. The decisions and actions controlled by KEEL Engines are 100% auditable, making them suitable for safety critical systems. KEEL Engines are small memory footprint functions that can be deployed in low level microprocessors. There is no need for costly high performance computing solutions. The potential exists to deploy KEEL Engines as hardware-only solutions making them suitable for "very" high performance systems (adaptive engine controls / smart bullets). The engines can be deployed at any hierarchical level.

A platform technology like KEEL Technology is needed to handle complex problem sets that must be addressed now in order to meet the needs of the future. Examples can be found in: weapon systems, financial systems, transportation systems, utility systems, etc.; especially those systems that must be both explainable and auditable. KEEL Technology also addresses the need to manage life cycle cost issues and be able to support the rapid development cycle that is required to address constantly evolving scenarios.

Whenever there is a need to model, explain, or deploy solutions to complex, dynamic, non-linear, inter-related, multi-dimensional problem sets KEEL can be used to add substantial value.

KEEL Technology supports these needs now. KEEL Technology is only available from Compsim.

Plans of Action:

Organizations with the responsibility to meet the needs for better and faster decisions should become aware of KEEL Technology in order to remain competitive. Once those organizations validate for themselves that KEEL Technology can provide a better mechanism for describing and executing complex, auditable behaviors (policies), KEEL Technology should be incorporated into collections of available technologies. This will increase capabilities and reduce life cycle costs.

References:

Rittel, Horst and Melvin Webber. “Dilemmas in general Theory of Planning” Report, University of California, Berkeley 1973

Tobin, Frank, “Modeling is Powerful, BUT Has Far to Go”, Bio-IT World, http://www.bio-itworld.com/pb/2009/01/22/tobin-modeling-letter.html

McCrone, John, “’Right Brain’ or ‘Left Brain’ Myth Or Reality?” The New Scientist; http://www.rense.com/general2/rb.htm

Clark, Admiral Vern; “Sea Power 21” Proceedings, October 2002; http://www.navy.mil/navydata/cno/proceedings.html

Andrus, D. Calvin “The Wiki and the Blog – Toward a Complex Adaptive Intelligence Community”, https://www.cia.gov/csi/studies/vol49no3/html_files/Wik_and_%20Blog_7.htm

Albus, James S. “A Reference Model Architecture for Intelligent Unmanned Ground Vehicles”, Proceedings of the SPIE 16th Annual International Symposium on Aerospace/Defense Sensing, Simulation and Controls, Orlando, FL, April 1-5, 2002

Compsim paper; “Knowledge Enhanced Electronic Logic for Embedded Intelligence” http://www.compsim.com/papers/About_KEEL.pdf

Papers, Demonstrations, Frequently Asked Questions on Compsim website: http://www.compsim.com