Extending Human Interoperability

I. Introduction:

It is a complex, competitive world. In order to compete, one is faced with a continuous flow of issues that need to be addressed. Human verbal and written language (the English language, for example) is commonly used to describe individual views of the world. When information is exchanged between individuals using this language it is often subject to interpretation by both the producer and the consumer of the information. Frequently there is a misunderstanding between the parties as one party interprets the information one way and the other interprets it another way. Outsiders (people not directly participating in the exchange) often debate the information, twist the concepts, and amplify misunderstandings.

For a more efficient exchange of information, the description of information must be more explicit, understandable, and easy to use. It must be suitable for electronic interchange and for electronic storage and retrieval.

Also, because one is dealing with complex issues, one must have a formal way of describing non-linear characteristics. Some examples are: issues with features that vary in importance under different scenarios, functional relationships that vary based on time and space relationships, scenarios that evolve as attributes change, issues that evolve from general interest to safety critical / mass casualty issues…

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

In order to compete in the global stage, there will also be a need to automate some of the human interaction in order to keep pace with countries that will shortly have a more educated skilled workforce than the United States.

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 information can be traced back to humans. KEEL is NOT “artificial intelligence”. The KEEL “graphical” language provides a platform that facilitates human thinking by allowing humans to “see” how their descriptions of problem domains would function in a dynamic environment. The KEEL language allows humans the ability to articulate complex models of how to interpret information without resorting to complex mathematical formulas. This is an iterative process as most humans have not had the requirement or the ability to do this in the past. When these descriptions are exchanged with others, it will be possible to investigate cognitive models in explicit detail to fully understand them.

II. Alternative Information Exchange and Processing Models:

The English Language

Written and spoken languages have evolved since humans have inhabited the earth. They carry with them almost no cost since humans have built-in mechanisms to create sound (mouth/voice box), hear sound (ears), create written material (hands), read written material (eyes), and interpret audible and visual information (brain). They have created inflections in their spoken language to add emotions in order to help others interpret their values. They have extended their languages to communicate those values.

Written and spoken languages used in daily life, however, are not explicit. They are always subject to interpretation. The costs (time and money) associated with interpreting common spoken and written languages are astronomical because they are processed by un-auditable human judgment.

Mathematics

Mankind has recognized the need for explicit modeling of information for centuries. From simple mathematics to advanced forms of propositional logic, humans continue to search for new ways to explicitly define information.

When humans built “machines” they immediately found that human verbal and spoken languages were not sufficient to describe qualitative information that could be understood and manipulated by machines. They found that size and shape could only be described with numbers.

While advanced mathematics has allowed the definition of complex models to be created, those models are not necessarily comprehensible by individuals without extensive training. This has forced domain experts to describe their problems / tentative solutions using the English language (for example) to the mathematicians where they formalized the description. The domain expert may never know if the resultant solution actually matched their original idea and/or problem.

Costs associated with creating complex mathematical solutions to subjective problems (“wicked problems”) limit this approach to a selective few application areas.

Conventional Computer Languages (Rule-based systems)

Since the advent of the digital computer (the von Neumann architecture), computers have been processing rules and numbers. 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 were and are fundamental to most automated solutions that are available today. Computer scientists have attempted to exploit the capabilities of computers to create forms of artificial intelligence.

As these techniques were challenged to address more complex, dynamic systems their costs to develop and maintain them escalated. One example is the fact that complex mathematics is required to address non-linear systems. There are extremely high costs associated with translating ideas into mathematical formulas, and then into software "code" before it can be tested. The resulting large monolithic computer programs are often so large that they cannot be easily understood. 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). Once the complex solutions are developed, the domain expert who had the idea (or the individual that has responsibility for the policy that is to be captured), has almost no chance of understanding how it was actually implemented. Extending complex, inter-related, rule-based systems has the potential of introducing "bugs" into the system that (in safety critical systems) has the potential of catastrophic errors.

Extensible Markup Language (XML)

One of the most significant advancements in information modeling and information exchange has been the development and use of XML and associated technologies. The use of XML DTDs (Document Type Definitions) and XML schemas (more detailed definitions) have allowed a formal categorization of data. Tags that are readable by both humans and computers allow information to be organized for archival and retrieval. Numerous tools are available to help humans package their information in XML structures. Information that has been packaged as XML can more easily be manipulated through structured databases. Different communities create their own DTDs and Schemas relevant to their specific problem domains.

Structured data by itself does not describe how it is to be used.

III. Knowledge Enhanced Electronic Logic

KEEL Technology provides a dynamic modeling environment. It incorporates a "dynamic graphical language" that allows one to define functional relationships that equate to "judgment and reasoning". Hidden from the user is an information processing engine that is created *as the user is defining and interacting with the model*. The explicit nature of the language and the processing engine allows complex models to be developed and tested before and after deployment.

A user (domain expert) interacts with the tool in order to stimulate inputs and observe how the 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 "engine". This interactive process helps a user define information in a manner that can be "measured".

The KEEL language is truly "dynamic". The language's ability to "show" the system processing 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. If one is focusing on extending human interoperability, one must provide a way to articulate how artifacts are valued in changing situations. The KEEL dynamic graphical language does this.

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. Then they observe the resultant 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 others. The systems evolve; getting more and more capable over time. In many cases people who are responsible for addressing complex problem sets need tools and techniques that allow them to develop and refine their own understanding of the problem sets. The interactive nature of the language allows them to test different concepts and make changes and observe impacts "immediately". It helps humans "think".

KEEL specifically targets applications that are expected to evolve as their operating environment changes. Humans are in control all the time. Every decision and action is 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:

It is difficult to share what you cannot understand. In order for humans to be able to exchange "knowledge" the information must be understandable. Conventional rule-based systems, probability-based systems, biological-based systems, and fuzzy logic based models do not satisfy all of the needs to address the complex, dynamic problem sets that exist now and will exist in the future. These systems will need to:

- Provide 100% auditable behavior
- Address dynamic, non-linear, inter-related, multi-dimensional problem sets
- Create systems that can easily be extended and modified
- Control life cycle costs

KEEL Technology will not eliminate all conventional solutions, but can easily be used to extend them with increased capabilities. Because KEEL-based cognitive engines are platform and architecture independent they can be integrated into almost any existing or new architecture.

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 that must then explain the solution to a software engineer so that they can then embed it in simulations or devices.

KEEL "tools" can be used to package the cognitive models as KEEL "Engines". These engines are provided as text files that are compatible with existing IDEs (integrated development environments), thus not eliminating existing tools that the user may be familiar with. The engines themselves 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).

A platform technology like KEEL Technology is needed to address complex problem sets that must be addressed <u>now</u> to meet the needs of future weapon systems, financial systems, transportation systems, utility systems, etc.; especially those systems that require that they be both explainable and auditable. KEEL Technology also addresses the need to manage life cycle cost issues and also to support the rapid development cycle that will be required to address constantly evolving scenarios.

KEEL technology should be considered for any application domain where there is a need to address complex, dynamic, non-linear, inter-related, multi-dimensional problem sets.

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

Plan 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 for use. 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

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

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

Compsim paper; “Characteristics of Innovation – A Review of KEEL (Knowledge Enhance Electronic Logic)”
http://www.compsim.com/papers/Characteristics_of_Innovation.pdf