Application of Knowledge Enhanced Electronic Logic (KEEL®) Technology for Multi-INT Association & Reasoning Algorithms

Objective

Discuss the potential of Compsim's Knowledge Enhanced Electronic Logic (KEEL®) Technology as a primary information fusion technology to address complex, dynamic, non-linear, inter-related, multi-dimensional problems that have been characterized in the solicitation to include "multiple reasoning, knowledge based, and heuristic algorithms against the temporal and spatial (time and space) association challenges". These are exactly the kinds of problems KEEL Technology was created to support.

Characteristics of the problems are identified below (**Characteristics of Complex Problems**).

Since KEEL Technology is platform and architecture independent it can co-exist with other more conventional approaches. In this manner KEEL cognitive engines can accept information from other sources (sensors, database, synthesized knowledge, or humans, etc). KEEL cognitive engines can also drive other information collectors or control systems. Since KEEL Technology is architecture neutral, it can operate autonomously in individual devices, or it can be distributed across the web.

Characteristics of Complex Problems targeted by KEEL Technology:

- Complex problems may require that many data items need to be considered.
- Individual data items in a complex problem domain may contribute differently to multiple parts of the same problem domain.
- Data item relationships in complex problem domains may be non-linear.
- Solutions to complex problems may require that many variables be simultaneously controlled.
- Data items in complex problem domains need to be understood both for the potential importance and for their instantaneous value (or instantaneous importance). (Time value of decisions)
- Complex problems may be dynamic or have dynamic components.
- Solutions to complex problems may require suboptimal solutions to component options.
- Complex problems are often composed of a number of component problems that must be considered before responding to higher level problems.
- Complex problems are often identified with conflicting goals.
- Components of complex problem domains often have goals that compete with higher level goals.
- Complex problems may have time and distance involvement.
- Complex problems may require the selection of the best "option" from a set of options, each of which may include the control of analog (relative) outputs.
- Solutions to complex problems may include the generation of events.
- Events initiated as a solution to complex problems can be relative.

- Components of complex problem domains may or may not be well understood (by human domain experts).
- Complex problems may contain components (data items) that possess some level of randomness.
- Another view of complex systems is that they are "self-organizing".

Technical Summary:

Compsim (Brookfield, Wisconsin) has created a new technology called Knowledge Enhanced Electronic Logic (KEEL®) technology that allows one to capture, design, test, deploy and audit human-like reasoning or judgmental decision-making. KEEL has been described as a new form of mathematics; one where functional relationships are completely defined graphically. The simplicity of this approach allows a domain expert (not a mathematician or software engineer) to create, test, and audit complex cognitive models. It is supported with a “dynamic graphical language”, that allows a domain expert to “see the system think”, or “interpret information and balance inter-related alternatives” while the algorithmic cognitive models are being created. The technology portfolio includes the dynamic graphical language for describing, testing, and auditing the reasoning process, a method for accumulating information, the architecture for a KEEL “engine” that can be deployed in a computer based device or software application, and the architecture for a KEEL “engine” deployed as an analog circuit. The KEEL “engines” are the functions or class methods that process the cognitive algorithms (depending on the target computer language selected).

KEEL can also be used in intelligent control systems: where the problem set includes dynamic, non-linear, inter-related, and multi-dimensional information items.

KEEL creates an explicit solution. By explicit, we mean that it equates to a formula that can be completely explained and audited. It is an "expert system" in that it requires a human expert that understands how information is interpreted and inter-related.

KEEL is not Artificial Neural-Net (ANN), Fuzzy, Bayesian, “conventional” Rule-Based or “conventional” mathematics based, although it can easily co-exist with these solutions when appropriate. KEEL is architecture and platform independent. The small memory footprint makes the technology suitable for embedded devices.

Compsim has developed suggested guidelines for complex information fusion algorithms.

Guidelines:

- **A methodology must be provided that allows the domain expert to define the information fusion algorithm with sufficient granularity so that it can be exactly translated into a form that can be explicitly executed by a device or software application.**

  The KEEL “dynamic graphical language” allows information fusion algorithms to be described as continuous curves that can be interpreted by a device or software application as if they were discrete formulas.

- **The methodology for describing the information fusion algorithm must support the efficient development of complex, non-linear scenarios.**

  The KEEL dynamic graphical language provides an easy way to document how information is to be processed without resorting to conventional formulas. Computer source code for the algorithms is generated automatically so time and expense associated with code development and debugging is eliminated.

- **The execution engine for the device or software application that will execute the information fusion algorithm must be suitable for embedded real-time operation.**

  KEEL Engines that represent the domain expert's design are high-performance, small-footprint functions. It is desirable that the same information fusion algorithm used in simulations can be used in devices. Modeling and simulation systems will become the test bed for actual devices.

- **The methodology must be completely understandable so it can be efficiently tested before deployment.**

  All functionality is displayed graphically. By tracing the wires one can see instantaneously how different and potentially conflicting data items are interpreted. The algorithm developer can stimulate the design at any time during the development process and “see how the system responds”.

- **Device or software application performance needs to be audited after deployment.**

  Services are provided as part of the KEEL Toolkit such that real world data can animate the dynamic graphical language so that decisions and actions can be traced to their cause and justification. In this way simulations can reflect reality.

- **The efficiency of the entire information fusion algorithm life cycle must be considered (design, test, deploy, audit, extend).**

The design of a KEEL-based system is interactive, while it is being developed. This allows it to be tested during the development process. Conventional code is created automatically, thus avoiding human typing errors. Deployment is handled by providing text files in the source code language of choice for easy integration into any IDE (integrated development environment). Auditing is provided with the ability to easily review the real-world interpretation using black box file reviews. Visually observing the importance of information and relationships allows complex scenarios to be reviewed with relative ease. Data is absolute so there is no human interpretation of the results required. Designs can easily be extended, by inserting new items and linking them into existing algorithms. There is no need to start over every time.

- **The methodology must be architecture independent so it can be deployed on a variety of platforms and in a variety of situations.**

  KEEL Engines are architecture neutral. They are simply cognitive functions (engines) created in the programming language of choice. The system architect has complete control over the marshaling of information and the scheduling of algorithm interpretation and execution. The same algorithm can be deployed in a simulation, emulation, in the device / software application itself, in a training system, and in a web based demonstration without re-engineering.

Compsim / KEEL Technology Background:

Overall Scientific and Technical Merit:

KEEL technology was specifically developed so human-like subjective decision-making could be deployed in real-time embedded devices and software applications. It was based on how humans integrate "supporting" and "objecting arguments" as a mechanism for interpreting the value of information. KEEL was developed after transferring this simple decision-making model (originally developed in Compsim Management Tools, a software tool for organizational decision-making) to complex systems where multiple decisions need to be addressed as part of a collective solution. With these problems there is non-linearity in how information from one part of the problem impacts other parts of the problem domain in different non-linear ways. KEEL has other attributes: completely explainable / auditable decisions, and small memory footprint. The problems addressed by KEEL technology are webs of information items with no real limits to the number of inputs and the number of outputs that need to be addressed by a KEEL cognitive engine. Another key aspect of problems that are addressed with KEEL Technology is the need to efficiently create the algorithms.

The development of KEEL Technology required the development of a "dynamic graphical language", a model for human-like reasoning, and a model for deployment in computer based devices as well as a model for deployment as an analog circuit. The development also necessitated the incorporation of numerous system engineering tools to construct more complex solutions from units of knowledge (KEEL Function Block Diagramming tools). The KEEL Toolkit that hosts the dynamic graphical language automatically evaluates complex models for stability during the design process. The

KEEL Toolkit is used to create the KEEL cognitive engines for a variety of platforms and writes the code for the engines automatically (C, C++, Microsoft C++ .NET, Microsoft C#, Adobe Flash, Java, JavaScript, Octave (MATLAB), Python, Visual Basic, VB .NET, VBScript, PLC Structured Text and others).

To "check" the code created by the KEEL Toolkit, Compsim developed a code testing methodology that is used to check complex designs.

To audit the cognitive models for validity, Compsim integrated the capability to animate the dynamic graphical language that allows humans to "watch devices think" and trap on certain decisions and actions.

Compsim has been granted patents for the basic reasoning model, the KEEL dynamic graphical language, and the deployment architecture.

Compsim has produced numerous technical papers, application notes and demonstrations of KEEL Technology, many of which are available on Compsim's website. KEEL papers have been presented at several technical conferences including the Phoenix Challenge, Sandia Laboratory Cognitive Conference, ISA, and IEEE High Assurance Systems Engineering Conference and the CITSA 2007 (International Conference on Cybernetics and Information Technologies, Systems and Applications) receiving a "best paper" award.

Openness of Solution

KEEL Technology is platform and architecture independent.

Compsim LLC is a technology company providing next generation cognitive technology for application in military, medical, transportation, industrial automation, governmental / business, and electronic gaming markets. Compsim licenses its KEEL® technology for use in embedded devices, software applications and for the Internet. The website is: http://www.compsim.com.

Compsim LLC
PO Box 532
Brookfield, Wisconsin 53008

(262) 797-0418