Application Note: Application of KEEL Technology in Asset Management Systems (9/15/2003)

Objective:

Product suppliers have developed more and more complex products. These have been integrated into more and more complex systems.

While products have increased in complexity, low end versions have been commoditized. This has led the product suppliers to move into a service oriented business model.
Because the low end products carry low margins and fewer customers demand the more complex products, the product suppliers have had to look elsewhere for revenue.

In some cases this has meant expanding their service model. The intent is to request "their fair share" of the manufacturing dollar.

While Product / Service suppliers continue to search for an expanding revenue stream, the product users are being asked to reduce costs.
Since systems are so complex, the end user has been required to seek costly service contracts from the product suppliers. Because the systems are so complex, only the product supplier can fix them.

The service model is appropriate for Product / Service suppliers as long as all products are equivalent.
However, if any product manufacturer can provide a solution where the combination of hardware/software and support is significantly lower, that company will have a considerable advantage.

Presently it is the support content that carries the most cost to the end user.

If any product / service supplier can greatly reduce the support cost, then that supplier can gather a larger product share.

The cost of Asset Management is aggressively pronounced by product / service suppliers as a cost sensitive approach to keeping the manufacturing facility up to date.
The product / service company will hold up reports showing how much the client has saved by outsourcing the Asset Management responsibility.

The product / service company marks up the manpower costs and leverages that manpower across several companies.

If the manufacturer could obtain the same level of Asset Management service, but without paying the product / service supplier for the manpower he / she had to staff, then the manufacturer would encounter direct savings.

The suggestion, in this Application Note, is that the PLC itself has the potential to take on the role of the service technician in many situations, and thus save the manufacturer from paying for continuous on-line asset management.

Market:

Enterprise Asset Management Market to Grow to $1.9 Billion by 2007 from ARC, Three Allied Drive, Dedham, MA 02026, 781-471-1000, Fax 781-471-1100, E-mail info@arcweb.com, Web ARCweb.com.
"Enterprise Asset Management Market to Grow to $1.9 Billion by 2007

Dedham, Massachusetts; June 20, 2003: The worldwide market for Enterprise Asset Management (EAM) including IT assets is currently at $1.6 billion and is estimated to grow at the Cumulative Annual Growth Rate (CAGR) of 3.1 percent reaching $1.9 billion in 2007, according to a new study by the ARC Advisory Group entitled EAM/CMMS Solutions Worldwide Outlook.

“The majority of the EAM market growth will come from Application Service Provider (ASP) Web-hosted solutions and industrial IT Enterprise Asset Management (ITEAM) opportunities,” according to Houghton LeRoy, ARC Director of Research and author of the study. He continued, “The EAM market has matured where software license sales are declining and customer service requirements are increasing.”

Factors Contributing to Growth

Customer service and support expectations in asset intensive industries has increased, requiring more consulting and continuous improvement services from suppliers. Companies that have seen good benefits are expanding EAM functionality to other groups and facilities. Companies with little or no payback justifications are looking for help establishing key performance indicators to measure ROI. Some are even outsourcing the problem to service providers who offer performance guarantees.

Several EAM suppliers are realizing the strong growth potential for outsourced services such as ASP Web hosting which includes maintenance, repair, and operations electronic procurement (eMRO). Web hosting is a very attractive and cost effective approach for asset management, electronic purchasing, and other enterprise applications. It permits faster implementation, simplifies support, integrates distributed operations, facilitates collaborations and reduces Total Cost of Ownership (TCO). Customers pay only for needed functionality and can accurately determine their costs.

IT organizations have managed office networks, servers, and PC workstations for many years. The proliferation of IT standards and equipment throughout the enterprise has forced considerable overlap in IT and EAM solutions. ITEAM will become a natural extension of many EAM solutions over the next five years, offering considerable opportunities for sharing best practices and emerging technologies. A one-solution approach will eliminate unnecessary duplication of many common functions, helping to reduce costs, increase productivity, and better share critical asset information. ARC expects to see more mergers, acquisitions, and consolidation of EAM and ITEAM suppliers.

TEAM Strategy Supports the CALM Model

In the search for stronger revenue growth, many suppliers have adopted a Total Enterprise Asset Management (TEAM) strategy that addresses asset management for all hardware and software assets from shop floor to top floor. This approach requires collaboration between operations, maintenance, financial, and IT groups within a company. It promotes sharing of best practices and important asset information for optimizing performance, availability, and quality within a large corporation. It lays the foundation for the Collaborative Asset Lifecycle Management (CALM) model for sharing the risks and responsibilities of asset management with partners, suppliers, and third party service providers during all lifecycle phases."

Business Drivers

Reduced Engineering Design Cost & Time

Increased Differentiation of Their Products

Increased Market Coverage

Improved Delivery/Logistics - Performance & Reliability

Improved Inventory Management -Reduced Cost & Space

Improved Internal Efficiency - Reduced Fixed Expense

Improved Access to Information

Buying Factors

Ease of Use

Time to Market

Flexibility

Extensibility / Scalability

Desire for Special Features

Performance

Options

Single Source for All Products

Complex Systems

Enterprise Connectivity

Trends

Product Availability

Quality

Prestige

Lowest Product Cost

Lowest Installed System Cost

Lowest Lifecycle System Cost

By-Products

Purchasing Environment

Expectations of the Future

Efficiency

Reduce or Eliminate Other Costs

Explainable / Auditable

Considerations:

The schedule for evaluating the data

When complex cognitive systems are being created, the designer needs to consider the time sensitivity of input and output data.

In human systems, humans may try to postpone decisions until all the relevant data is available. When pressed, however, humans can react to demands that drive them to make instant decisions. In a KEEL system, the designer may need to have the system "adapt" to changing demands.

The flexibility of the KEEL system design allows for normal conservative operation and still adapts to emergency situations.

The system designer needs to consider when the data should be evaluated. Options could be to trigger the evaluation on a scheduled basis; it could be polled by a higher level authority; it could operate on a change of state from one of its sensors; or it could be continuously running.

Human interaction with the solution

In cases where KEEL engines participate with humans as part of an overall system, consideration should be given to the human interface characteristics of the system.

KEEL engines can trigger the dialog, or the dialog can be initiated by the human.

KEEL engines can provide continuous analog output which can be translated into human readable form or as stimulation to HMI tools.

KEEL engines can accept input from any source, as long as it is translated to the normalized format required by the KEEL engines.

The system designer will consider how and when the dialog between the human and the KEEL engine will be scheduled.

The ability to create complex cognitive solutions in a timely manner

Using the KEEL toolkit, it is possible to develop complex cognitive solutions in a short timeframe.

Software features are built into the toolset that allow the cognitive design to be structured in a manner that makes it easy to integrate the KEEL engine(s) into a variety of architectural models. In many cases, these tools allow the glue logic to be created once and be able to support changes to the cognitive model.

The KEEL-FBD tool allows the staged development of complex models in a manner that automatically creates the glue logic. This is another feature that supports rapid development of complex systems.

Performance of the system
Different applications require different levels of performance for the KEEL engines.

Some real time control applications may require that information is constantly being evaluated and operated upon.

There will be other times when information changes relatively slowly and therefore does not merit constant evaluation.

There are a variety of techniques to balance performance and system complexity when creating KEEL based solutions.

Architecture Independence
KEEL engines are discrete information processing engines.

In many cases they will be part of a larger system.

Just like humans in a factory, in an army, or on a team, there is interaction between individuals and between devices and equipment. Computer protocols and network architectures have been developed in a manner that somewhat duplicates the development of human language.

People developed different languages to satisfy different needs: voice for people to communicate verbally; sign language for those that cannot hear; telephone and video conferencing for distance communication.

Computer based protocols have been developed to satisfy different needs: ASCII for simplicity; XML for structure; asynchronous / synchronous / isochronous for flexibility, speed, and time determinism.

KEEL engines can participate with any of these linkages, as long as the data can be normalized to meet the needs of the KEEL engines.

KEEL engines can sit at any point in any architecture. In this way KEEL engines can be integrated into almost any architecture. The simple API enables this valuable attribute.

The evolution of the KEEL engine over time
KEEL engines are often developed over time. New pieces of information are added to the design when it is apparent that they contribute to the interpretation of the data provided.

Compsim's KEEL Toolkit provides a number of services to enable enhancements to the engine over time without impacting the glue logic. This saves software development effort.

The KEEL FBD tools allow a complex system to be developed in stages and integrated and tuned as separate components.

The KEEL toolkit incorporates the idea of merge-able objects or decision-making modules.

There are other cases where the model doesn't need to expand, but only needs to be tuned when relationships between information change.

KEEL engines can be created as "classes" in some languages.

The "importance" of information
A key aspect of any KEEL based system is the dynamically changing importance of information.

In simple systems, there may be no changing importance of information. These types of systems might be built with hard coded solutions or discrete logic.

Cognitive solutions, however, usually deal with complex relationships where information is being interpreted in different ways; in different parts of the same problem.

Cognitive problems often deal with both strategic (future) and tactical (now) problems at the same time. Questions about what to do now and not destroy future opportunities are often addressed. This requires a balancing of information to obtain the best outcome.

Diagnostics and prognostics require that a system adapts its operation to performance variables. If you have difficulty breathing, then breathing becomes more important.

The value of "explainable actions"
Many systems can benefit from an engine that creates explainable decisions and actions.

This allows the systems to be audited for their performance in order to tune them over time.

It is possible to have KEEL engines monitor other KEEL engines and potentially provide a feedback mechanism to achieve optimal operation.

There are other cases where there are demands for the creation of explainable actions. This insures that a code of ethics is integrated into the design. Without the ability to decompose every action and every decision, there is the ability to create a system that does not meet the needs of society.

So, KEEL based systems, because they are rule based, can explain why any action was taken or decision was made.

Even though the rules are defined graphically, by inserting a snapshot of the inputs back into the design, the reasoning can be displayed.

Who and how the KEEL actions are monitored
KEEL based systems offer several methods for monitoring their performance.

First, because all KEEL actions are visible and explainable in the development environment, they are available for analysis within the tool environment.

Second, because a KEEL engine is a rule based system, it will always respond the same way to an input. An XML schema exists that defines the format for an XML file produced by the real-world device or software applications. If the device or software application logs the input data in this format, it can be read back into the KEEL Toolkit where the reasoning can be reviewed and explained.

The system designer has the responsibility for determining when and how the data is logged and reviewed.

Certain applications may demand more auditing to insure that performance is satisfactory.

It is also likely that new data will enter the application space. This may trigger reviews of performance.

As in human systems, novice operators or players may require closer review than experienced operators or players. The same is true with a KEEL engine.

Consideration for how the KEEL engine(s) fits in the "chain of command"
In some cases KEEL engines will participate as components of a larger system.

They can perform administrative roles by interpreting information in a consistent manner and responding with consistent command decisions.

They can perform subservient roles by accepting direction from above and adapting the commands to modify their actions.

They can sit in the middle of a chain, by accepting commands from above and reviewing status from below. They can modify their own strategy according to the rules provided to them and the information they observe on their own.

They can make requests to humans and devices above them in the chain-of-command, and can deliver commands to humans and devices below. They can collaborate with their peers according to the rules that dictate responsibility.

Should it be appropriate, the KEEL engines can develop their own levels of trust in collaborative environments. The system designer will determine the communication protocols and the flexibility of the system.

The level of "trust" attached to input information
Many judgmental decisions are made by including a level of trust to validate the information. When the level of trust is diminished, then the information may be given a lower level of importance in the overall solution.

KEEL engines can include the level of "trust" as an input to the system. How this is interpreted, is the responsibility of the system designer.

The concept of risk associated with the decisions and actions associated with the system
Many cognitive decisions need to incorporate risk into the decision-making model.

Risk can be included as an input to a KEEL engine.

It is the responsibility of the system designer to determine how risk participates in the decision-making model.

How time and space relationships might contribute to the solution
Time and space often impact the importance of information when making cognitive decisions. KEEL supports these concepts with its "clipper" features.

This allows decisions and actions to be tuned for different times and locations.

In cases where an optimal solution is being targeted, such as the time to send a message or the time to shoot at a moving target, then tools to support these kinds of decisions are required. They are built into the KEEL toolkit and they are available to the system designer.

Normalizing the data

KEEL engines normally operate on normalized data (0 to 100) values that can be either integer or floating point, as defined by the KEEL project. Any normalization scheme can be used.

While this suggests a linear range of normalized input data, the inputs can drive a curve which allows the data to be interpreted in almost any way.

In this manner a single normalized input value can be interpreted according to any number of independent curve relationships.

Much of the development work in architecting a KEEL solution is spent defining the relationships between information. Because this is all done graphically, there is no need to write "code" to see the results of the analysis.

The responsibility for the overall system remains with the solution architect

The system architect is responsible for the overall architecture of the system.

This will include segmentation of the system, determining when and how the KEEL engines will be scheduled.

The cognitive model for interpreting the input data and causing decisions and actions to be promoted is also the responsibility of the system designer.

System architecture

The system architecture is the definition of the relationships between all system components.

The cognitive segment is usually just part of the system. The system architecture defines the layout for performance, flexibility, extensibility, cost, resources, etc.

The system architecture is often the result of a balancing act: balancing time to market, resources, performance, and cost. The system architecture is driven with an objective where the features are defined. The objective is addressed by identifying potential solutions: selection of components and methods of tying them together.

KEEL technology can be integrated into the architecture from the beginning, or it can be an "add-on". Because individual KEEL engines are architecture neutral, they can be integrated into an overall architecture at different times; even after a program is completed.

The potential for autonomous operation

Because KEEL engines can interpret information in a human-like manner, and direct relative actions to be taken based on that interpretation, KEEL based systems have the potential to operate without human intervention.

Alternatively, KEEL based systems can operate as either backups to human operators, giving advice or recommendations, or they can provide the primary decision-making engines that are backed up by humans.

The outputs from the system

The outputs from a KEEL engine are normalized values between 0 and 100. This information may have to be transformed into other forms for use by the external controls or monitoring equipment.

KEEL could generate control signals.

KEEL engines interpret information and provide outputs that represent a balancing of the inputs. These values can be used to generate complex commands in the form of control signals to other equipment.

KEEL could generate information for other KEEL engines.

A common practice is to segment a system into multiple KEEL engines. It is likely that one KEEL engine will provide data to the input(s) of other KEEL engines.

The KEEL FBD tool assists in connecting KEEL engines into a single solution.

KEEL engines could also be distributed across a network or in multiple tasks where messaging or data sharing could provide the mechanism for one KEEL engine to feed others.

KEEL could provide inputs to other systems (non-KEEL).

The outputs from a KEEL engine could be supplied to other non-KEEL subsystems for further processing.

KEEL outputs could be part of a local feedback loop.

KEEL engines can be part of a feedback loop, where the output of the system is connected back to the input through some other circuitry.

KEEL outputs could be part of a distributed feedback loop.

KEEL outputs could be fed to other external devices which, in turn, feed data back to the input of the KEEL engine. The other devices could be local or remote to the KEEL engine.

Warning messages could be triggered.
The outputs from the KEEL engine could be used to trigger warning messages. The warning messages could use other outputs to describe the warning in relative terms.

Information messages could be triggered to indicate status. Analog values could be included to explain subjective interpretation.
KEEL engines can used to supply variable information in the form of informational messages. Complex messages could be structured from multiple variable output signals.

Commands to operators could be generated.
KEEL engines interpret information and provide outputs that represent a balancing of the inputs. These values can be used to generate complex commands to an operator.

Logging of information could be triggered. KEEL could log its own decisions or it could log other inputs and outputs.
The outputs from KEEL engines could be logged for historical records or for audits.

The inputs to the system could also be logged. If the log format is in XML compliant with the KEEL Input Schema, then the data could be used in the development environment to recreate the decision-making model for exact interpretation.

KEEL could cause state changes of the system to take place based on subjective evaluations.
KEEL engines interpret input information and drive outputs. These outputs could drive an external state machine that could cause the equipment holding the KEEL engine (or any other system with or without the KEEL engine) to change state. In this case, the KEEL engine is supplying inputs to the state machine.

KEEL could generate diagnostic interpretations.
Beyond just generating processed / interpreted information, KEEL engines can interpret diagnostic information and explain the interpretation in detail.

In addition to explaining the interpretation, it can provide the information to explain why other diagnostic interpretations are not considered.

The sources of input data
KEEL engines can accept inputs from almost any type of data source, as long as the data can be transformed into the normalized data format required by the KEEL engine.

When the input is textual or verbal, it will have to be transformed into the normalized format.

Sensors
Sensors of all types can be used as inputs to a KEEL system. As long as the information can be transformed into one or more normalized input values, it can be interpreted by a KEEL engine.

Sensors exist to detect and measure almost all physical states. For example: time, temperature, pressure, torque, speed, acceleration, distance, density, color, edges, shapes, counts, volume, etc. There are probably sensors to measure anything for which a value can be assigned to it.

Collections of sensors can also detect and measure non-physical information: stress, truthfulness, pain. These values are determined with some algorithm that synthesizes the information.

Human operator
In systems where the human operator is part of the system there is the potential that the operator will be providing input data to the system.

For example, the operator could be supplying input data to the system as part of the job function. The operator could be reading values or recording physical observations about characteristics of the problem domain. This could be a doctor that records physical symptoms of the patient or of the environment that may contribute to the symptoms.

In other cases, the operator could be directed to take specific measurements. An example might be an automotive service technician that could be directed to take readings in an automotive electrical system to try and isolate the problem.

Inputs from human operators are commonly gathered through some kind of Human Interface Device that will transform the information from human terms to formats more conducive to digital processing. This might be via a keypad, a pushbutton, or in some cases it might be voice input. It could also be in some form of visual form where information is generated by physical movement. It is possible that any of the human senses could trigger inputs to a KEEL system.

Databases
Databases are used to store historic and synthesized data. This data can be manipulated by any number of mathematical processes to provide running averages, identify trends, detect shifts, etc.

The result of database queries can generate numeric information that can be used as inputs to KEEL engines.

Databases that are constantly updated have the ability to send evolving data to KEEL engines and thus tune the KEEL engines with new data.

External Data Sources
In addition to databases, external data sources can be any device or software application that generates information.

For example, machine tools may have counters embedded in them that count completed operations or completed orders. This information is gathered as the equipment operates and can provide input information to KEEL engines.

A clock or calendar is another example of an external device that can generate information for a KEEL engine. These devices generate time related information.

A communication network might generate traffic information.

Other KEEL Engines
KEEL engines may be components of a larger cognitive system. In these cases it is likely that one KEEL engine will feed other KEEL engines.

The KEEL FBD tool provides a mechanism for integrating multiple KEEL engines in a single application.

A more loosely coupled solution would be to connect KEEL engines across a network or some other connectivity approach.

Locally accumulated data
A KEEL engine will be embedded in a device or software application.

It is likely that the device or software application will be performing functions in addition to the cognitive process associated with the KEEL engine. In these cases other values generated by the application may be used as inputs to the KEEL engine. Certainly diagnostic and prognostic data generated by a device might be used to drive a KEEL engine.

Preprocessed data

Preprocessed data can exist anywhere in a system. This preprocessed data could have gone through a validation process or a transformation process. It could carry with it confidence data or some other biasing information that could be used in conjunction with the preprocessed data. It could be accumulated locally or it could exist anywhere in the system where it could be move to the KEEL engine for interpretation and processing.

Control Signals from Other Devices

Control Signals from pieces of equipment or software applications can be used as inputs to KEEL engines.

In some cases KEEL engines are part of autonomous devices. They react to their surroundings and decide what to do for themselves. In some of these cases, the KEEL engine could intercept control signals from another device that are directed to perform operations for that other device. This information could provide intelligence for the device containing the KEEL engine.

Connectivity

The ability to install KEEL engines at any point in an architecture makes this a valuable attribute where components of the system are likely to be distributed across different pieces of equipment in different locations.

KEEL engines can be connected by any media and in any format as long as the data is converted to the normalized format before triggering the KEEL engine to process it.

Directly wired to source

In its simplest form a sensor can be directly wired to an input pin on a microprocessor where the signal is transformed to a normalized data format used by the KEEL engine.

The same is true for the output. In its simplest form the normalized data output from the KEEL engine is transformed before sending it out a pin on the microprocessor where a wire carries the signal to an actuator (control point).

A direct wire is the simplest form of network.

Network connected - any topology

Connectivity to and from KEEL engines can be accomplished with any type of network with any topology.

Some wired networks might be termed point to point, multi-drop, token passing, star, web, loop, trunk, etc.

Messaging techniques might include: Broadcast, Store and Forward, Directly Addressable, Group Addressable, All-call.

Any message packaging technique might be used: structured or unstructured, packed, block mode, etc.

Any character encoding can be used: ASCII, Async, Bisync, Isochronous, or any other.

The data can be encrypted or non-encrypted.

The choice of network connectivity is left to the system designer.

Infrared connection
The KEEL engine is not restricted to any specific connectivity to its inputs and outputs. Infrared links can be used.

Supplied by the same processor running the KEEL engine
Because some data sources and data sinks for KEEL engines will be within the same microprocessor, input and output data can be generated and consumed locally.

Radio Frequency
KEEL engines are independent of the connectivity between inputs and outputs and the KEEL engine. Radio frequency connectivity is appropriate for some applications where wired and infrared connections are not appropriate.

Description:

Define the expectations (outputs) from the system.

Identify the sources of information.

Define the information hierarchy which includes how and when information is accumulated.

Determine where KEEL engines might be located in the system and what information will be exchanged.

Plan for staged introduction.

Evaluate KEEL segmentation.

KEEL General:

This section identifies the general benefits that could be derived from a KEEL based solution.

Human experts are required to interpret information to make the best decisions or take the most appropriate actions

Devices can make control decisions when human operators are not present

Repetitive judgmental decisions are prone to error

Applications where the judgmental decisions must be explained

Complex situations where it is uneconomical to develop and maintain straight line code (IF, THEN, ELSE)

Situations where the environment is dynamic and the importance of information changes and the system must react to change

Where architectural issues may prohibit other solutions (KEEL technology is architecture independent: localized, distributed, web based, multiprocessor…

Summary:

An Asset Management system with products that can automatically adapt to changing demands can create new market for hardware devices.

Products with the human expertise ready to take corrective actions without service technicians around can greatly reduce their dependency on support personnel and therefore the costs associated.

Product vendors with the ability to integrate the Asset Management functionality into their products will have the potential to cause a market shift in their direction as the overall cost to the customer is lower.

The Product Vendors will also be able to increase their margins if they have a competitive advantage.

Disclaimer
This application note suggests the potential for KEEL technology to respond to certain market needs. The end users are totally responsible for assuring that the technology performs as expected.

The application note may also assume that certain external technology exists to support the KEEL engine in an effective manner. This may or may not be accurate in all cases.

