

Risks, Requirements Based Technology Management & Results

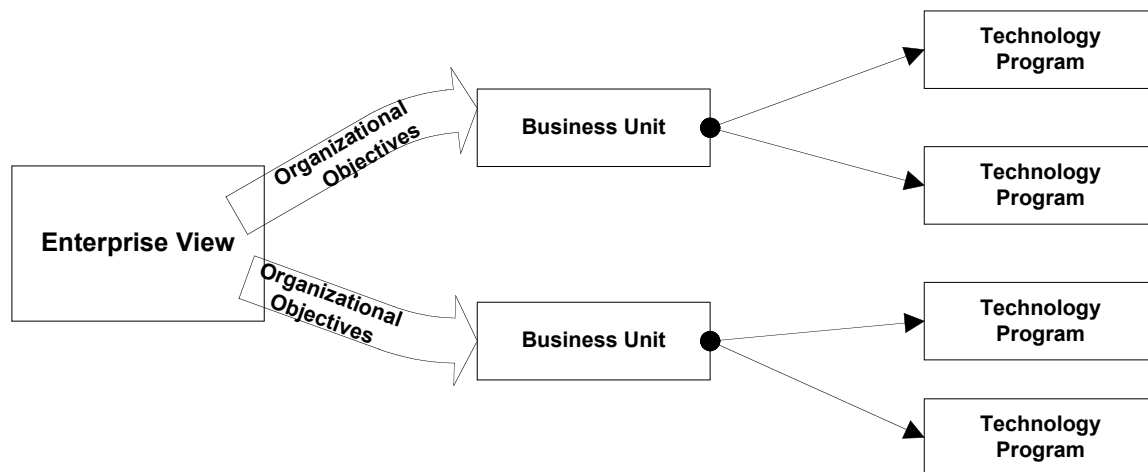
Organizations spend significant amounts of resources managing knowledge. How does an organization measure its success or status of activities in meeting its goals? Does it take a “system of systems” viewpoint or treat each individual technology program as a standalone?

SONEX Enterprises Inc, a successful Northern Virginia small business firm, believes organizations must take a “Systems of Systems” viewpoint and requirements centric management approach.

A requirements centric management approach applies controls for an enterprise view and individual program/project view. Requirements Based Technology Management means an organization understands that requirements can and will change over a product lifecycle. The key is how quickly can an assessment be made to measure the impact of changes.

SONEX was established in 1984 with a single focus: to adapt the Systems Engineering approach to the integration science for evaluating organizations and systems through Independent Verification and Validation (IV&V).

To concentrate on this objective, SONEX developed its proprietary Integrated System Assurance™ (ISA). ISA is a business process implemented through web-based technologies. SONEX’s innovative ISA concept, consistent with industry quality standards, introduces a repertoire of cost effective tools to enhance the success of organizational objectives and individual technology programs. This process assures decision makers at all levels with near real-time oversight and assessment of all organizational technical, life cycle and program management activities.



Enterprise View - ISA™ enables organizations to track enterprise objectives and the impact of issues on those objectives. The enterprise can be further segregated down to business units, divisions and/or branches. At the base of the enterprise view are the programs or projects that support the organizational objectives.

Program View – Each program or project has its own detailed evaluation hierarchy segmented by phase to provide near-real time status of technical, cost and schedule issues. Also, ISA establishes relationships between programs so that decisions made in one program can be quickly evaluated as to their impact on dependent or interoperable programs. This is a critical capability. Many times a seemingly “simple” change in one program may have major cost and schedule implications on another. ISA™ enables the seamless transfer of information between program decision makers .

The ISA™ foundation resides in consensus establishment of the dominant quality factors for each view, Enterprise and program. That is, stakeholders establish a set of 4-6 evaluation criteria to provide a common basis for issue analysis. Management evaluates any issue against these criteria to use resources against issues that impact the quality model and not expended on those that have lesser value. ISA™ reduces risk by focusing on evaluating the selected quality factors while being sensitive to the objective functions that define their relevance and balance over the schedule. Our processes reinforce and are consistent with ISO 9000 and other industry standards.

SONEX ISA Process

The SONEX Web-based ISA process allows distributed teams and/or decision makers to share information and view the status of issues and objectives in near real-time. The tools described below are not a product you purchase, but are part of the service we provide.

SONEX developed a suite of tools to facilitate the ISA™ business process. Historically, processes, such as ISA, usually require intensive labor by expert analysts. Recognizing these characteristics, SONEX initiated a tool development effort in 1990 focused on fielding automated tools specifically designed to minimize the time and skill level required to implement the ISA™ business process.

Exploiting Artificial Intelligence technologies, specifically in the field of Expert Knowledge systems, SONEX successfully developed highly sophisticated rules-based tools to enable trained systems analysts to work as if they were experts. SONEX's analytical tools are integral components of SONEX's *EXpert Test ENvironment for Development (EXTEND)* architecture. This open tools environment provides a framework for building the specialized applications needed to perform ISA™ functions.

Government research and development organizations, universities and industry partners all endorse the SONEX EXTEND tool development. The U.S. Army's Communications-Electronics Command (CECOM) funded much of the EXTEND tool development effort through Small Business Innovative Research (SBIR) grants. George Mason University and the Northern Virginia Center for Innovative Technology provide graduate students and additional funds to advance the research and development of the EXTEND tools.

Most software engineering tools assume that the functional requirements are correct - a very misleading prospect for the systems developer. The tools developed under EXTEND, such as the REQSPERT and RT2 (see below) are based on the assumption that *the requirements may be wrong!* Therefore, the EXTEND tools focus on testing,

identifying and correcting requirements errors much earlier in the life cycle than any other product currently available. This approach produces the agreement we are looking for before the developer starts designing and coding.

The Requirements Expert (REQSPERT)

REQSPERT is comprised of several modules. In the operational mode, sets of system requirements are entered, scanned, or transferred into the tool via ASCII format.

REQSPERT:

- Classifies the requirements interactively
- Conducts a conflict analysis (to include identification of voids)
- Assigns tools to test each requirement (requirements testability analysis)
- Identifies COTS tools to be employed
- Develops test plan items

REQSPERT's flexible report writer provides problem tracking reports, tools lists and the technical segments of a proposed test plan, together with the assumptions made to derive them. Organizations can perform all of these functions now - even before a contract is awarded to develop a target system.

The Requirements Tracing Tool (RT2)

SONEX developed the Requirements Tracing Tool (RT2) with the support of the U.S. Army (CECOM). RT2 supports developing and maintaining requirements traceability through all phases of the system's life cycle. As with other components under the EXTEND architecture, RT2 is designed to significantly reduce the time, effort, skill levels, and ultimately the cost required to establish an independent trace of requirements from functional specification through design, code and testing.

Why would organizations establish and maintain traces?

When a technology or product is ready for release, what controls verify the end product is what the enterprise desired? RT2 provides such a report so decision makers have the confidence level the product meets the enterprise business objectives.

RT2 exploits expert system technology to provide an automatic means of identifying probable relationships between documents, for example, between enterprise policy and program requirements. RT2 associates a confidence level with each probable trace. An analyst, therefore, faces a smaller subset of items to evaluate for possible tracing, and this trace can be verified by confidence level. The knowledge bases to determine relationships between requirements and specifications are flexible and can be trained to support any system domain with minimal effort.

REQSPERT and RT2 provide efficient requirements baseline discipline and provide unique capabilities to manage and mitigate risks.

The Requirements Tracing Tool (RT2) supports requirements traceability from enterprise objectives down to software code or system components.

The Intelligent Agent Definition Toolkit (IADToolkit)

The IADToolkit provides the mean to prototype, through evolutionary simulation scenarios, the messaging rules of a rule-based information distribution systems (RBIDSs). The approach taken to design and implement such a tool kit is based on a software architecture of a set of agents and modules. A description of each module is as follows.

- **Problem-Scenarios Agent**

At a user's request through a graphical user interface (GUI), the problem-scenarios agent generates problem scenario information as well as the environment information at which the problem scenario occurred. The problem scenario information contains the initial state of the environment at which the problem scenario has occurred, the sequence of external events which caused the problem to occur, and the results that would have been expected if no errors were to occur. The expected results are the required information at each individual node based on the requirements specification.

- **Test-plan Module**

Based on the problem-scenario information as well as the environment information, the test-plans module generates a relevant test plan. A test plan is a sequence of test cases. Each test case is a scenario that is a potential problem scenario. These test cases are used to test the existing rules as well as the generated rule sets to achieve a certain level of confidence. All the test cases have the same initial state but they will have different sequences of external events and different goal states. The problem scenario itself may be included as one of the test cases. If the IDRs are exhaustively tested with a test plan that contains all the possible test cases, then confidence level of *100%* is achieved. Of course as the number of possible events in the environment gets larger, Exhaustively testing the IDRs with all possible test cases will be impractical. The test-plans module has a reasoning capabilities that minimize the number of test cases while maximizing the confidence level.

- **Adaptation Module**

Once the test plan is generated, its test cases are passed to the adaptation module for the purpose of removing errors as well as optimizing the IDRs. The adaptation module is a debugger.

- **Fault-detection Module**

Since there is a chance that the test-plan will not be able to exhaust all possible events, any set of rules that has not been exercised during the correction process will be a potential faulty set with faults. To test for faults, the modified optimized rule sets are passed on to the fault-inspecting module (which is a fault-detection module). The fault-inspecting module gets, as an input, rule bases and inspect them for redundancy, contradiction, etc. If a fault is found, the fault-inspecting module generates a fault indication as an output. The fault indication is received by the adaptation module for the purpose of removing the fault. The cycle between the adaptation module and the fault-inspecting module will continue until no fault is to be found. Once there is no fault to be found in the modified rule bases, those rule bases are passed on back to the problem-

scenarios agent as an optimal rule sets for the given problem scenario and other possible similar scenarios.

The IADToolkit reflects a continuation of SONEX's current research and development efforts to build Independent/Integrated Systems Assurance (ISA™) rule-based tools. This tool provides an excellent means for validating system rules and managing changes for sophisticated rules-based information systems. The tool can also be used for configuration control by providing developers capabilities for anticipating the effect of a component maintenance activity on the overall software system.

Cost/Schedule Control System (C/SCS)

SONEX ISA™ supports most project management software such as MS Project. SONEX is capable of loading data from customer activities along the critical path of a specified program, granulate the Work Breakdown Structure (WBS) to the 7th or 8th level and analyze the data to assess the probability of a developer meeting its proposed schedule. Similarly, SONEX can assess developer recommendations for incremental schedule changes and assess their true impact on the total program schedule. All customer tasks will be incorporated into ISA™ and their respective status will be tracked, analyzed, and reported through this system.

Results Instead of Risks

Organizations and customers look for results not risks. Systems support more and more critical business functions. Meeting the enterprise objectives and getting systems right, and right the first time, has never been more important. SONEX can help both the organization and the program manager get it right.