

Situation Assessment and Decision Making

Integrated into the Process Centered Environment

Stephanie White, Susan Dorchak, and John Keane, Long Island University and System World Inc.
William Pallack, System World Inc.
Jerry Owens, IntelliTek
Jerzy Rozenblit, University of Arizona
James Davis and Janos Sztipanovits, Vanderbilt University

Abstract

Researchers have defined a number of process modeling methods and have developed in-roads to process-centered environments that support process modeling and project control. However, there is little research that incorporates variability of the human condition into process modeling. The negative effects of numerous variables on the quality of human situation assessment and decision making can be as detrimental to expected results as any catastrophic failure.

The research documented in this paper proposes a model for Situation Assessment, and derives a taxonomy of individual, task, environmental, and organizational attributes that can affect situation assessment and decision making.¹ The model and attributes are integrated with a state-based process modeling paradigm. The MultiGraph Architecture, Vanderbilt University's model-integrated computing environment, was used quickly and cost-effectively to generate a graphical process modeling environment that accounts for situation assessment and decision making.

1: Introduction

Corporations want to improve and automate their processes to increase system quality, customer satisfaction and profit margin. Forward thinking companies are modeling their processes to better understand and automate them. They need capabilities that do not exist today, and which are rarely addressed in process research. They need methods that model

variability in human activity due to factors which are internal and external to the human agents. Industry needs these methods because processes are human oriented [1], and "interactions among humans and between humans and the tools that support their activities are characterized by high variability and unpredictability" [2].

Our objective in developing a Process Centered Environment (PCE) is to support humans who are engineering complex systems to achieve the objectives (both process and product) for which they are responsible. The term "process centered environment" means a framework and collection of software modules that provide expert aid for modeling processes and for tailoring tools to support the Project using that process. The purpose of the PCE is to promote good design by enhancing the human engineering process, overcoming human limitations, and fostering user acceptance. The PCE should perform the following functions, adapted from Rouse [1].

- Automate the engineering process as needed but keep the user in charge;
- Support browsing and linking of on-line sources of process information;
- Retrieve process text and graphics from all sources;
- Support the creation, editing, execution and evaluation of process simulations, both qualitative and numeric;
- Support training in the engineering and product development process, tools and methods;
- Perform technical administration (track design decisions, check process parameters versus constraints, etc.).

2: Process Modeling Paradigm

Our process modeling paradigm consists of a core paradigm derived from known process modeling methods,

¹Work funded by NSWC and ONR under SBIR Contract Number NOOI 78-97-C-30 17

combined with a model of situation assessment and decision making. A detailed report on our research is documented in [3].

The core paradigm models activities, resources, products, and product flow.

- Activity behavior is modeled as hierarchical state machines, as in Harel's Statecharts [4]. This method has been shown to provide excellent support for process modeling [5],[6].

- Resource models define human, machine, tool, and facility resources. Resource behavior is defined using hierarchical state machines. Individuals (agents), skills, organization elements, and non-human resources are assigned to activities. The Organization view of the resource model is networked and supports the Integrated Product Team (IPT) approach.

- The Product aspect of the Activity Model defines all products of the process, such as system and support items, services, knowledge, and documents. This aspect also defines the number and type of elements that constitute each product, and specifies alternate elements where required.

- Inter-activity product/information flow is defined in the Product Flow model, which is based on the Dataflow model. Knowledge, documents and other products are output from activities and are used by other activities. Since human resources are assigned to activities, model analysis can reveal the degree and content of communication among human players.

Our research integrates a Model of Situation Assessment and Decision Making with the core paradigm. The notional "situation model" is adopted here as a

convenient way to reference an individual's state of knowledge concerning the various elements in a dynamic environment. In Figure 1, a situation model is explicitly represented as equivalent to situation awareness and is assumed to be produced as a result of the situation assessment processes. Situation assessment (an activity) leads to a situation model (situation awareness) that can be defined as an artifact. Decision making (an activity), then proceeds to produce a decision (an artifact). Figure 1 depicts important activities (i.e., perceive, comprehend, project future states) involved in situation assessments that are conducted on the part of the human (agent), the activity of decision making, and the decision artifact that can lead to specific actions. The model in Figure 1 is consistent with assumptions made in several theoretical foundations, for example [7],[8], and [9].

The number of independent variables that could possibly affect situation assessment and decision analysis is obviously large, and accurate a priori predictions of the effects of multiple variables (and their interactions) on human information processing activities is less than straightforward. Variable effects might be linearly combined to yield a hypothesized overall effect. In actuality, effects would likely combine in a non-linear fashion that might be modeled by weighting the contribution of each effect, taking into consideration the simultaneous presence (or absence) of other effects. Speed and accuracy in situation assessment and decision making can be assumed to be negatively or positively affected based upon the presence, absence, and magnitude of effect of the variables.

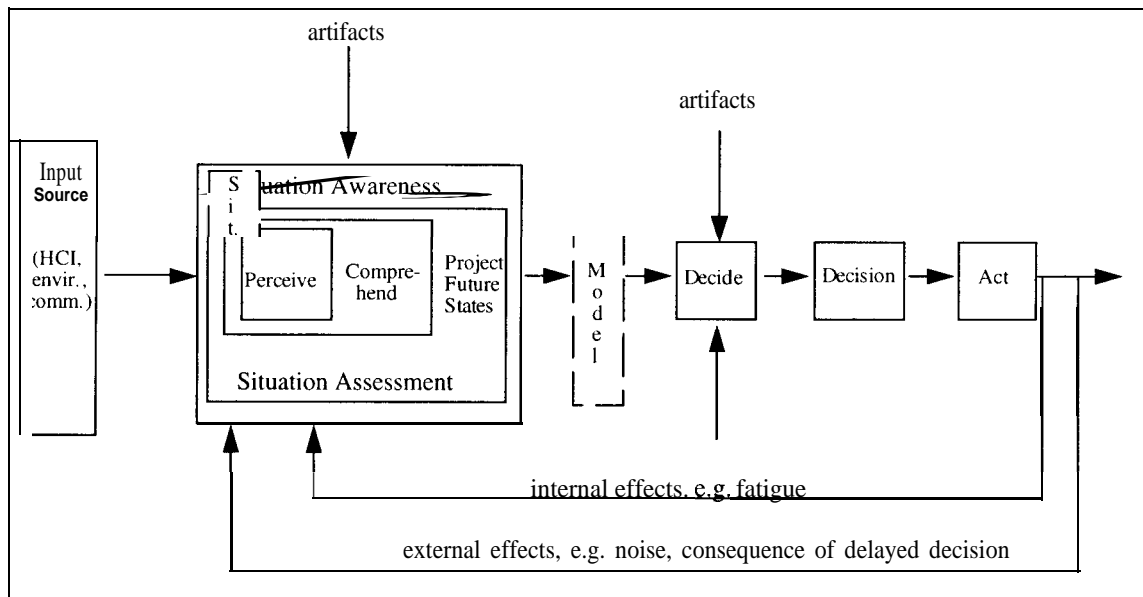


Figure 1. Model of Human Situation Assessment and Decision Making Processes.

A threshold value can be established that must not be exceeded if the activities of situation assessment and decision making are to proceed. Operational assumptions and conditions at the beginning of the simulation can be used to establish initial states and corresponding values for the variables that can be selectively “tuned” (altered) as the simulation proceeds and conditions change, possibly in response to the introduction of new constraints.

Four categories of variables that can affect an individual’s situation assessment and decision analysis capability are:

- (1) Individual (Physical/Psychological),
- (2) Task/System,
- (3) Environment,
- (4) Organization/Social.

An individual’s physical variables include fatigue, sleep loss, and work-rest cycle demands. Psychological variables include lack of experience, lack of knowledge/expertise, lack of confidence in information, lack of flexibility/adaptability, lack of risk management ability, lack of communication skills, and lack of motivation to perform. Task/System variables include task criticality demands, task workload demands, time stress demands, cost (time, fiscal, human resources) to acquire additional information, consequence of delayed decision, consequence of no decision, consequence of error, and poor information displays. Environmental variables include distractions such as noise lighting, heat, and cold. Organization/Social variables include cost containment demands, schedule demands, regulation/policy demands, lack of manpower/team members, consequence of decision on others, lack of team training, and lack of team coordination /communication skills.

Team situation assessment and decision making in the context of engineering design can also be influenced by numerous factors. The individual, task-system, environmental, and organizational/social variables proposed to affect individual situation awareness and decision making are relevant here also as individual performances combine to determine the quality, utility, timeliness, etc., of the output artifact or product. Other factors are implicated in the team situation, however, as individual responsibilities are often interdependent and performances are inter-linked. Foremost among these factors are the experience of the team in working together and the communication among members regarding their individual and collective assessments of design issues and progress toward meeting specific and overall design objectives.

In the taxonomy presented above, individual design experience and communication skills were taken into account. Experience in working together as a team also may be viewed as a variable and one that might be improved as a function of the frequency and duration of interactions within a team setting. The act of

communication, (as distinguished from skill in communication), however, holds the most promise for devising an evaluation framework to account for situation assessment and decision analysis on the part of an engineering team. In particular, communication outputs (artifacts) can provide the basis for measures of effectiveness of individual and team situation awareness and decision analysis.

3: Example

Engineers make numerous decisions during system design and development about requirements correctness and completeness, design-tradeoffs, test. sufficiency, make/buy, and so on. We recommend modeling these important process decisions, and their inter-dependencies. This includes modeling the information to be perceived and comprehended, and the projections that should be made for each decision, see figure 1.

Figure 2 shows the hierarchical state machine for making the decision, “Ready to Design?”. The team projects the impact of insufficient requirements on design quality and cost, whenever management asks if the requirements are sufficient to proceed. In order to make this projection, the team must comprehend the criticality of requirements that are not defined, and risks that will be incurred. To understand these potential problems, the team has to do a thorough analysis of where requirements may be inconsistent, incomplete, or incorrect. Information flow for the “Ready to Design?” process would be documented in a Dataflow Model.

Variables that affect Situation Assessment and Decision Making are modeled as state machines. These variables can be considered artifacts, as they result from the individual, task, environment, or organizational base and activity. The states “High”, “Medium”, and “Low” represent the magnitude of the negative effect (high negative effect, etc.) that an artifact such as workload demand has on overall Situation Assessment and Decision Making. For the activity, “Projecting Impact of Insufficient Requirements on Design Quality and Cost”, workload demand depends on the number of requirements issues to investigate and the nearness of the deadline. Artifact effects such as workload in concurrent activities, fatigue, noise, lack of experience, or lack of information might be combined to yield a hypothesized overall effect. Effects would likely combine in a non-linear fashion that might be modeled by weighting the contribution of each effect, taking into consideration the simultaneous presence (or absence) of other effects. Speed and accuracy in situation assessment, and making the decision as to whether the team is ready to design the system, can be assumed to be negatively or positively affected based upon the presence, absence, and magnitude of effect of the variables/artifacts.

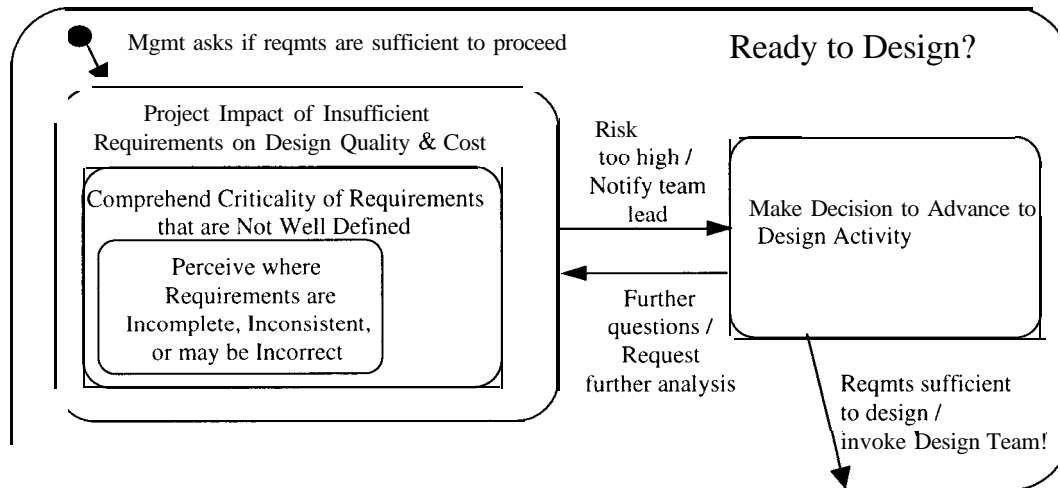


Figure 2. Situation Assessment & Decision Making Represented in the Process Model

One way to account for the combinatorial effect of artifacts/factors affecting human agents in situation assessment and decision making is to model process behavior in such a way that certain activities are performed only when threshold values are within an acceptable range.

The factors that affect Situation Assessment and Decision Making, such as task workload demand, have been modeled as state machines with three values: “high”, “medium”, and “low”. However, it might be preferable to model these factors using other mechanisms such as Abdel-Hamid and Madnick’s equations involving dependent variables [10],[11] or Zadeh’s fuzzy sets [12].

Abdel-Hamid’s work may integrate well with our concepts. Abdel-Hamid represents over one hundred interdependent phenomena affecting the software development process, and expresses the dynamic interdependencies of these phenomena. However the model is defined at the macro level, and does not model individual activities, information flow, organizations, and products.

Zadeh’s theory of fuzzy sets may be a good candidate for mechanizing the effects of factors, internal and external to the human agent, on situation assessment & decision making. The concept provides a natural way of dealing with problems in which the source of imprecision is the absence of well defined criteria of class membership rather than the presence of random variables.

4: Use of MultiGraph Architecture for Tool Development

In *model-integrated computing*, integrated, multiple-view, domain-specific models capture information relevant to the system under design. Models explicitly represent the designer’s understanding of an entire system. Integrated modeling explicitly represents dependencies and constraints among various modeling views.

The Multigraph Architecture (MGA) is an infrastructure for model-integrated computing and is described in detail elsewhere [13].

The integrated environment includes Modeling Tools, Integrated Model Database, Analysis Tools, and Application Synthesis Tools. The Analysis Tools work with tool-specific analysis models; the applications are specified in terms of executable models. The modeling paradigm of the analysis tools and the executable models are domain independent. In a given domain, the relevant information about the design artifact is captured by a multiple-view, domain specific modeling paradigm. In MGA, the modeling paradigm is described by a meta-language. The meta-language representation of the modeling paradigm is used to generate components of a Metaprogrammable Model Server, and a Metaprogrammable Graphical Model Editor (META-GME). A Model Interpreter is the key to tool integration in MGA. The role of the Model Interpreters is to translate the domain specific model into analysis models and executable models. This architecture allows analysis and synthesis tools to share design information that is common without requiring that the tools use the same modeling paradigm.

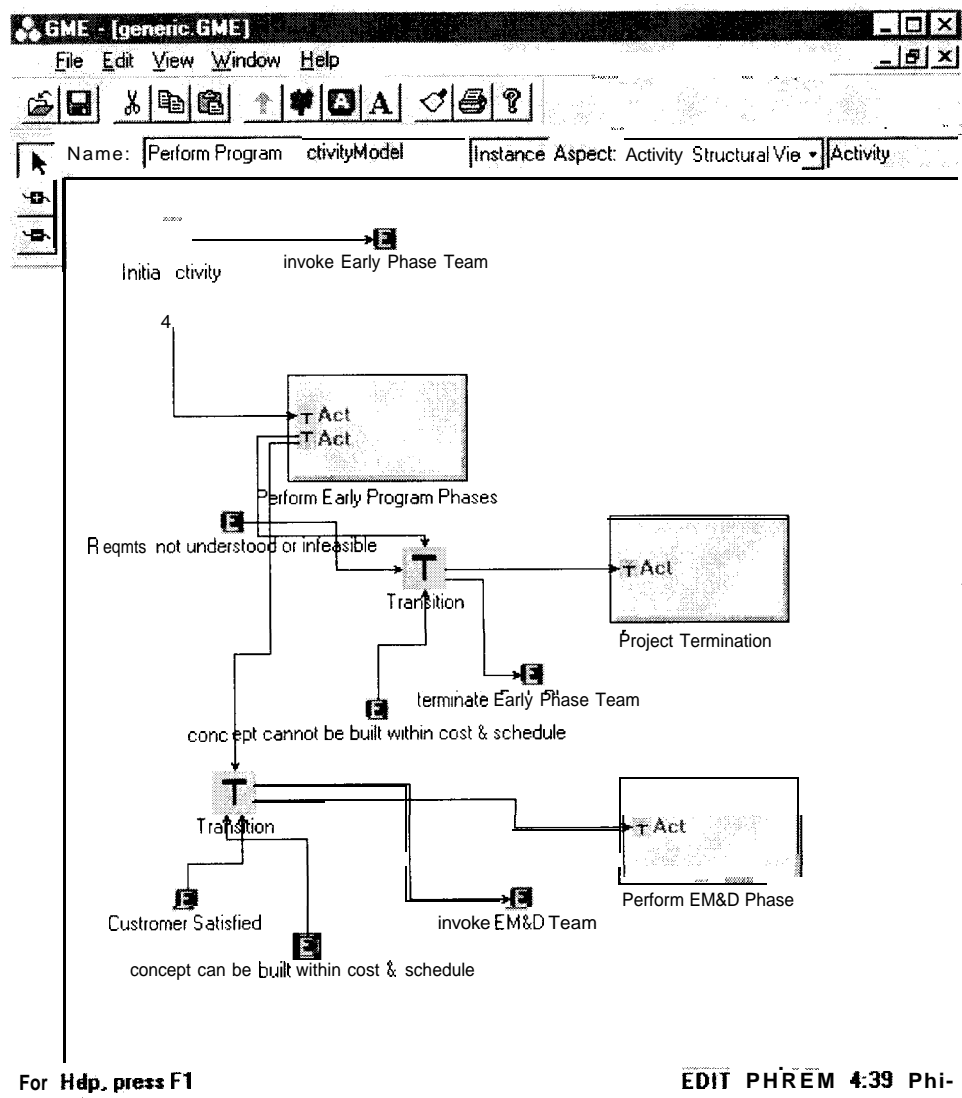


Figure 3. FSM Modeling with Multigraph Graphical Model Editor (GME)

5: Process Modeling with Multigraph GME

Process Centered Environment (PCE) models are comprised of Activity Models, several types of Variable/Artifact Models that affect Situation Assessment and Decision Making, and Resource Models. Each of the models and model interaction is described below.

Activity Models. The Activity Model, which is the heart of the PCE modeling paradigm, is used to describe the different activities (or processes) that comprise the modeled system. The FSM model is the main Activity decomposition aspect. An example Finite State Machine (FSM) model in GME is shown in Figure 3.

The semantics defined in [4] are used to interpret the FSM models. States may consist of other states, transitions, events, parameters, input transitions, output

transitions, state references, and initial state specifications, as well as references to events, parameters, input transitions, and state references. Input and output transitions are used to connect states. States contain decomposition attributes that allow the user to specify whether the sub-states are parallel, independent states or sequential states. The FSM models used for modeling activities contain references to events and states of Variable Models. Thus, changes in values of Variables (Individual Physical/Psychological, Task/System, Environment, and Organization/Social) can affect Activities. The Product Flow View specifies information/product flow between Activities. Activities can also be assigned a risk, which can be used in risk assessment/mitigation analyses.

Activity Models also contain a Fault View aspect that models the activity when special events (*fault*

events) occur. Using special transitions, the Activity Model's nominal behavior is augmented with the fault behavior. These models are then used in risk analyses to determine what steps can be taken to mitigate the risks. The Vanderbilt State Space Analysis Tool (SSAT) can be used for this type of risk assessment.

The Product Aspect defines the type of product, such as system items, support items, services, knowledge, and documents. The type and number of each element required to produce the product is modeled, as well as alternative elements that can be used.

Activities also contain a Resource Utilization aspect. This aspect is used to model which resources the Activity requires and the amount of time/utilization the Activity requires of the Resource. The Resource could be an individual person, a team of people, a piece of software, a piece of hardware, or even a facility.

Variable/Artifact Models. Variables that affect quality and performance in Situation Assessment and Decision Making are modeled using FSM models. All Variables have three states (high, medium, low) representing the magnitude of negative effects the Variable can have on an Activity.

Resource Models. Resource Models represent individual as well as groups of resources. These models

(1) Can be hierarchically decomposed to show organizational views of the Resources.

(2) Can contain references to other Resources, which allows the representation of one Resource to be a member of multiple teams or organizations,

(3) Contain a Performance Aspect, which shows how Activity Models can affect the Performance of a given resource, and

(4) Contain references to Resource Behavior Models, which are modeled using FSMs.

Analysis Tools for the PCE Environment. The MGA tool interface technique has been applied to a wide variety of problem domains. It has proven to sufficiently integrate analysis tools with a modeling environment. Vanderbilt University's past experience leads us to believe that currently available analysis tools can be incorporated into the PCE environment. To date, we have only used the Graphical Model Editor to support process modeling. In the future, we hope to use a number of other tools developed at Vanderbilt University to perform consistency, completeness, reachability, fault tree analysis, and formal analysis of the state space. As analyses are identified as desirable in the PCE environment, analysis tools that perform the required analyses can be developed and integrated. As other analysis tools are identified, the PCE modeling paradigm will have to be examined and possibly modified. Then, a model interpreter can be written to interface the analysis tool to the PCE environment.

6: Conclusions and Recommendations for Future Work

Our goal is to use the Vanderbilt University Multigraph Architecture as a framework for a process-centered engineering environment. The approach allows the fast and inexpensive prototyping of key ideas, and can serve as a foundation for building a complete toolset.

The process models we develop using the Vanderbilt Multigraph Architecture will be used for:

- Generating and executing process simulations,
- Analyzing processes,
- Situation Assessment and Decision Making,
- Customizing support tools for the different phases of the process, and
- Integrating a complete tool environment supporting the modeled engineering process. The integrated environment will include customized tools for process workflow, measurement of process status based on process metrics, causal analysis, and others.

Rozenblit's domain engineering approach to Simulation Based Design will support process tradeoffs and process integration through capture of process knowledge, creation of process taxonomies and process variants, and definition of process coupling rules. Knowledge representation schemes will be used to capture, classify, organize, and manage knowledge about processes, equipment, performance criteria, and so on [14].

One of the major objectives of the present work was to go beyond a strictly deterministic modeling approach to process engineering, by taking into account important idiosyncratic and non-deterministic aspects of human activity. An improved understanding of the roles and functioning of multi-disciplined engineering teams is viewed as essential to the larger goals of being able to apply robust process metrics in tailoring workflow, which will lead to more cost-effective products and services. The research produced a descriptive framework that identified critical variable sets that can affect human situation assessment and decision making in a process engineering team environment. The framework also included an agent (human) processing model depicting situation assessment and decision making and the way that artifact (variable) effects can map onto these agent activities. Actions resulting from agent decisions were then related to design criteria, requirements, and goal satisfaction that, in turn, could be shared with an Integrated Product Team Leader and other engineering team members. Effective communication was identified as one of the most important determiners of team success and a method to represent and assess the utility of structured communications among team members has been proposed. The framework to account for human activity

was developed to be consistent with the current process modeling paradigms, to be integrative and applicable within a real situational domain, and to be portable across problems and domains.

A number of issues remain to be addressed in demonstrating the utility of human activity modeling in support of simulation-based process engineering. The descriptive framework developed in the present research exemplifies the role of agent situation assessment and decision making in process engineering. Techniques are required to demonstrate how the human activity modeling assumptions can be enacted in a dynamic simulation environment to support decision analysis on different levels. At least two levels of analysis seem pertinent to our present goals. The first level of analysis concerns the effects of various combinations of internal (agent generated) and external (task/environment generated) variables/artifacts on decision making performance. A second level of analysis concerns the specific trade-off decisions an engineer must make often in the face of a number of competing constraints, alternatives, and goals. In the first case, the global effects of the workplace environment and the agent's performing role in situation assessment and decision making are of primary interest. In the second case, the focus is upon a particular decision required in response to an immediate problem with a number of defined dimensions. The descriptive framework took into account important constraints and activities involved at both levels of analysis, but a method for analyzing decision making efficiency and effectiveness in the context of overall workplace constraints and in relation to a specific problem remains to be derived.

Fuzzy sets represent a promising avenue for extending our work in human activity modeling. Human situation assessment and decision making are often conducted under conditions characterized by uncertainty, vagueness, and smoothly changing values. Even the simplest decision can be quite complex in view of the numerous inputs or constraints that may need to be considered almost simultaneously. Humans are very adept in their ability to "weigh" the inputs to a problem and to make decisions even though it is impossible for them to consider all possible input combinations before taking action. Real-world systems present fuzzy situations for human decision making and the resulting decisions likewise can be described as fuzzy. The methods of fuzzy logic provide an ordered way of capturing all the possible combinations of ranges of inputs in rule-based structures. Human activity modeling for process engineering should be compatible with many of the assumptions of fuzzy sets as dimensions (inputs) of interest (e.g., constraints, goals, alternatives, and constraint/goal/alternative importance and satisfaction criteria) can be specified in relative (fuzzy) terms. The value results associated with alternatives could be used at the global level of analysis to prescribe changes in workflow, modify procedures,

adjust constraints, etc. At the trade-off decision level of analysis, alternative value results would identify desired courses of action for the decision maker.

7: References

- [1] Rouse W. B., *Design for Success, A Human-Centered Approach to Designing Successful Products and Systems*, John Wiley & Sons, New York NY, 1991, ISBN 0-471-52483-2.
- [2] Finkelstein, Anthony, Kramer Jeff, and Nuseibeh, Bashar, eds. *Software Process Modeling and Technology*, John Wiley & Sons, New York, 1994.
- [3] White, S., Pallack, P., Dorchak, S. Keane, J., Sztioanovits, J., Davis, J., Rozenblit, J., and Owens, J., *A' Process Centered Environment for Complex System Development that Emphasizes Situation Assessment and Decision Making*, SBIR Report for Naval Surface Warfare Center, Contract No. C-N00178-97-C-3017, 1998.
- [4] Harel, D., "StateCharts: A Visual Formalism For Complex Systems", *Science of Computer Programming* 8, pp. 231-278, 1987.
- [5] Armitage J.W., Briand L., Kellner M.I., Over J.W., and Phillips R.W., *Software Process Definition Guide: Content of Enactable Software Process Representations*, Special Report, CMU/SEI-94-SR-21, Software Engineering Institute, Carnegie Mellon University, Pittsburgh PA, 1994.
- [6] Raffo, D.M., *Modeling Software Processes Quantitative\ and Assessing the Impact of Potential Process Changes in Process Performance*, Ph.D. Dissertation, UMI Dissertation Services, Ann Arbor MI. 1996.
- [7] Endsley, M. R., "Toward a theory of situation awareness in dynamic systems." *Human Factors*, 37(1), pp.32-64.
- [8] Harris, S. D., Owens, J. M., Barrett, C. L., Parisi, M. A., & Becker, D., "Sensor fusion and situation assessment: A perspective", *Proceedings of the 4th National Symposium on Sensor Fusion*, 1991.
- [9] VanDijk, T.A. and Kintsch, W., *Strategies of discourse comprehension*, New York: Academic Press, 1983.
- [10] Abdel-Hamid, T. & Madnick, S., *Software Project Dynamics: An Integrated Approach*, Englewood Cliffs NJ, Prentice-Hall, 1991.
- [11] Abdel-Hamid, T.K., "The Slippery Path to Productivity Improvement", *IEEE Computer*, July 1996, pp. 43 - 52.
- [12] Zadeh, L.A., "Fuzzy Sets", *Information and Control*, 8, 338-353, 1965.
- [13] Sztipanovits, J., et al., "MULTIGRAPH: An Architecture for Model-Integrated Computing" *Proc. of the IEEE ICECCS'95* Ft. Lauderdale, Florida, Nov. 6-10. 1995.
- [14] Rozenblit, J.W. and C. Kocourek, Concepts for Computer Assisted Engineering Process Management, Proceedings of the 1997 IEEE Conference and Workshop on Engineering of Computer Based Systems, 405-412, Monterey, CA, March 1997.

Email: swhite@panix.com; dorchak@hornet.liunet.edu; jkeane@eagle.liunet.edu; dwdm66a@prodigy.com (Pallack); jowens125@aol.com; jr@ece.arizona.edu (Rozenblit); bubba@vuse.vanderbilt.edu (Davis); sztipaj@vuse.vanderbilt.edu (Sztipanovits)