

Billman, D. (2014, March). Work Representations for Evaluating and Modeling Human-Machine Systems. In 2014 AAAI Spring Symposium Series.

<http://www.aaai.org/Library/Symposia/Spring/ss14-02.php>

<http://www.aaai.org/ocs/index.php/SSS/SSS14/paper/view/7758>

# Work Representations for Evaluating and Modeling Human-Machine Systems

**Dorrit Billman**

San Jose State University

Dorrit.billman@nasa.gov

## Abstract

Effective modeling of Human-Machine systems depends on an appropriate representation of work. This note discusses why format of work representation is important for modeling. Work representations vary in how specific versus abstract they are, with different levels providing different benefit. I identify some limitations of the predominant emphasis on highly specific work representations and benefits of pursuing more abstract representations. A new, abstract representation format (Work-Function X Variable Matrices) is proposed and illustrated with an example from cockpit aviation; a contrasting, specific-level example is sketched.

## Introduction

Formal analysis methods depend on the representation of the situation being modeled. Model verification methods for Human-Machine systems (H-MS) depend on the representation format used in the models and on how the model represents the target Human-Machine System. Many Human-Machine systems are intended to support complex work and analysis in complex work domains depends critically on the representation of the work the technology is intended to support; both the representation format and the content represented for a particular problem situation are important yet difficult to develop. An important goal of H-MS modeling projects is to ensure that the technology modeled is suitable for the work it was intended to support, namely, that it is fit-for-purpose.

*Work representation*, here, broadly covers any representation of work, including tasks, constraints, functions, and ontologies. Different types of work representation are likely to be most helpful in different modeling contexts, for different types of work, at different stages of development, and for different evaluative scope. I suggest that existing work representations used for modeling Human-Machine

systems have focused primarily on specific, procedure-oriented representations, and that it is valuable also to develop more abstract representations. I propose a framework for considering appropriateness of different work representation formats for modeling, describe one example of a relatively abstract representation, and sketch a second contrasting example. The level of abstraction of a model of a human-machine system is influenced by the level of abstraction of the work representation on which it is based.

## Concepts & Scope of Application

*Computer* refers to any sort of computational device, whether stand-alone or embedded, general or special purpose, and its associated software. *Sociotechnical system* refers to a combination of human and computer elements that work together for some purpose and that is, at least in part, designed for this purpose (aka human-machine system). *Automation* refers broadly to the capability of a computer to carry out activity without human involvement, typically involving actions in the physical rather than (only) the informational realm and typically actions which have alternatively been done by people.

The scope of applicability of our proposal is modeling of sociotechnical systems that share the following broad characteristics. 1) The sociotechnical system critically relies on computation, decisions, and judgment. Broadly this implies some level of complexity and openness. 2) The system is concerned with work. That is, the activity has extrinsic, stable, trans-individual goals, constraints, and consequences. 3) The system or component of interest is identified at a level where humans and computers interact. This contrasts with completely automated activities (e.g., the timing of spark-delivery to a car engine) and with completely human activity (e.g., some medical interviews). A focal concern is with the functionality provided by an engineered system (i.e., *computer*) to a worker. The work

here is primarily motivated by application to safety critical systems, particularly those with extensive automation.

Effective design and evaluation of sociotechnical systems with respect to support of the work-needs requires a characterization of those needs. Poor representation of the needs to be addressed contributes to system failures and accidents (Charette, 2005; Ellis, 2000; Pew & Mavor, 2007). Good representation enables assessing the match between the problem posed by the intended function of supporting the work and the solution offered by the system being developed. Note that a representation of work-needs, which is a problem representation, is logically prior to a traditional requirements specification, which is a solution representation. Realistically, capturing all relevant aspects of work within one representation type is unlikely; multiple complementary representations may be valuable. Understanding how different representations complement one another should enable more useful modeling.

### **Work Representations: Abstract vs. Specific**

Representations of work vary in how abstractly they represent the work. Representations that are very abstract characterize work at a high level, in terms of goals, missions, broad constraints, or very abstract tasks. Representations that are very specific characterize work at a low level, in terms of specific goals, particular action sets, or “button-pressing” procedures. Related notions of abstraction level are wide spread. Vallacher & Wegner (1987) point out the multiple levels of description of intentions that people offer, from trying to get together with someone, to pressing the button of the doorbell. Kieras (1997) discusses the concept of high- versus low-level task analysis. The construct of means-ends hierarchy taps into a related dimension (Naikar, Hopcroft, & Moylan, 2005; Vicente, 1999).

Specific representation formats have predominated in Human Computer Interaction, cognitive engineering, and Human-Machine modeling. Traditional HCI methods emphasize evaluation at a specific level and focus on usability of a particular design rather than assessing functionality relative to work needs (Card, Moran, & Newell, 1983). Traditional task analysis also emphasizes specific characterization of tasks, assuming the more detailed the better and that it is primarily the expense that limits detail in the level of description (Kirwan & Ainsworth, 1992). RAFIV analysis (Fennell, Sherry, Roberts, Jr., & Feary, 2006) also characterizes the step-by-step processing a user must do. Work in formal modeling also relies on specific, procedural models to trace out the consequences of action (Bolton & Bass, 2009; Palanque & Bastide, 1997). Highly specific representations support scenario-based modeling, and running samples of particular behaviors in a specific design.

Abstract representations focus on identifying general, necessary characteristics or needs of the work domain. The most familiar approach favoring abstract representation is Cognitive Work Analysis (CWA). CWA began in part as a reaction against highly proceduralized work methods, and with skepticism about including specific procedures as requirements on design or evaluation (Rasmussen, Pejtersen, & Goodstein, 1994). Kieras’s (1997) high-level task analysis emphasizes the benefits of moving away from highly specific representation.

More abstract representations of work encourage new solutions that may differ from any existing solution by meeting the abstract needs in new ways; they enable comparative evaluation across different systems and for high-level, underspecified designs; they allow broad coverage where work activity is relatively open rather than pre-specified. Highly abstract representations, however, are unlikely to provide sufficient detail to ensure a human-machine system is well designed. Design choices not represented at an abstract level may prove critical. Highly specific representations focus on providing a detailed problem characterization. Though it may be hard to ensure coverage of work needs by relying on specific, scenario-oriented representations, evaluations based on specific work representations can provide assurance that the solution is sufficient (though not necessary) for the represented work needs. One adequate solution among many might be confirmed. This approach is conservative, potentially admitting only as small class of viable solutions, of potentially low quality. Representations at higher levels of abstraction are more likely to be structural rather than procedural, abstracting away temporal information. Representations at lower levels of abstraction are more likely to specify temporal or sequential information, as in procedures.

### **Work Characteristics: Generative vs. Routine**

The benefit from different levels of abstraction in representing work depends on characteristics of the work. The *target-work* is the bounded body of work intended to be supported by some technology. Target-work areas differ in how generative versus routine the work is. Routine work can largely be solved by reuse of previous solutions, either those actually encountered or those anticipated in design. Generative work requires productive combination of small solution components to generate novel solutions.

Any target-work area includes a distribution of work elements, varying in how generative or routine each is. Several factors affect how heavily a target-work area is skewed toward generative vs routine work. If the target-work area creates a large problem space, with many possible combinations of circumstances, actions, and outcomes, it is likely to have a higher proportion of generative work

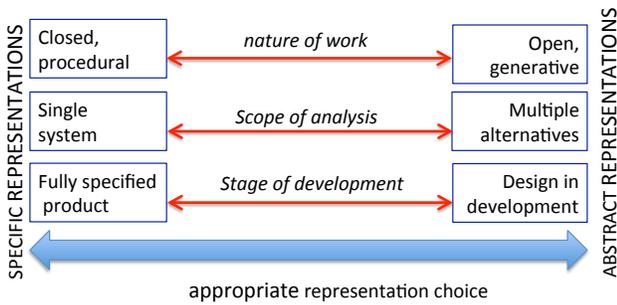


Figure 1. Factors affecting abstraction level of work representations

within it: fewer of all the combinations are likely to have been encountered in practice or imagined in design. Conducting one medical test or designing a bolt has a smaller problem space than conducting a large and diverse suite of tests or designing a house.

For any given target-work area, its problem space may have been more or less intensively explored, driven by factors such as its cultural importance and cost of error. For example, the work of flying an airliner and of air traffic control each have very large problem spaces; many aspects of this work have, however, been intensively explored both through numbers of flight hours and through systematic, experimental investigation. In contrast, target-work areas that may in fact be much smaller, perhaps troubleshooting and diagnosis of a one-of-a-kind life-support system in the International Space Station, are much less explored and hence may remain less understood and more generative.

In addition, the degree to which a target-work area is routine versus generative is influenced by the way it is affected by external elements. Target-work that is relatively insulated from external perturbations, whose boundary is relatively impermeable, can be more routine, since the contextual affects are limited and do not substantially increase the size of the problem space. Further, the degree that such external perturbations are known is also very important; while aviation is strongly affected by weather, most of the impacts are understood and routine responses have been developed for many. Perturbations such as a tsunami's impact on a power plant, which are extreme and unanticipated, result in the need for highly generative work.

Target-work areas differ in the degree to which they allow routine or require generative work, and also in how well understood the boundary is between the two types. More 'meta-level' difficulty is introduced when it is hard to determine whether or not a routine solution is applicable.

Intrinsic complexity, degree of cultural exploration, and ease of identifying when an old solution applies all shape the distribution of routine versus generative work required

for effective performance in that target area. These factors shape the distribution of performance choices available to the cognitive agents (human or computer), e.g. across Rasmussen's skill, rule, and knowledge level (Rasmussen et al., 1994).

## Factors Affecting Level of Work-Representation

Figure 1 shows three factors influencing what abstraction level of work-representation is appropriate. The nature of a target-work area is important. While I believe abstract work representations are widely, they are critical for target-work areas that have a high proportion of generative work. Here a given objective might be accomplished in many ways, so describing work at a very specific level may "force work into a straightjacket". Specific work representations will be useful for work-areas with predominate elements of routine work and also for work-areas in which the routine work is well demarcated. Here it may be appropriate to specify elements of work at a highly specific, proceduralized level.

Purpose of analysis influences the most useful level of abstraction. The purpose may be to evaluate multiple possible systems, perhaps to compare the degree and loci of inadequacies; then the work representation should not characterize how work is done specific to one system. If no system yet exists when the work representation is formulated (as in cases of evaluating design) an abstract representation is required: details of how to do the work do not yet exist. If the purpose is evaluating a single, in-use system, a detailed specification of the work may be helpful.

Finally, scope of the target-work to be represented may be an important factor. It may be infeasible to represent a broad scope of work at a highly specific level, or an abstract representation may be required to organize a decomposition into subdomains which are feasible to represent.

### High-Level Representation: Cockpit Example

The Work-Function X Variable Matrix work representation is a new, relatively abstract framework for representing work. It selectively synthesizes function-oriented aspects of high-level task analysis (Kieras, 1997) and the idea of a census of domain variables from CWA. It abstracts away many aspects of work such as temporal structure and means-ends relations.

Work-Function X Variable representation was developed in the context of representing complex automation, specifically a target-work area within cockpit aviation. This target-work area has an interesting balance between generative and routine work. It has a large problem space, but much has been intensively explored. Frequently,

though not always the boundary from routine to generative work requirements is clear. A large proportion of work is closely structured by detailed procedures and policies. However, unanticipated need for generative recombination of solution components is encountered in complex, multi-fault or extreme off-nominal situations, and is a critical part of a pilot's work. A key application goal is to compare and evaluate alternative designs for pilot-automation integration. Thus, a critical requirement is representing work at a sufficiently abstract level to allow comparing how well alternative designs support the work. Thus work functions are abstract such as "accept an Air Traffic Control Directive", and well above the "button pushing level". To make development and application of the work representation tractable, our initial analysis only includes nominal situations, while "providing hooks" for representing off-nominal cases. The target work area specifies standard flight and autoflight capabilities of a passengerliner but does not assume anything about the interaction resources available.

This Function X Variable representation characterizes work as a set of *work functions*, a set of *variables*, and their associations. A *work function* is a high-level, goal-focused task or activity intended as part of the work (e.g. accept Air Traffic Control (ATC) directive to change altitude). Work is decomposed into partially separable functions that collectively accomplish the mission within constraints. Each *function* is characterized at a sufficiently abstract level to avoid assumptions about the technology being designed or evaluated; the work representation is independent of the technology variations it might be used to assess. For example, functions concerning communication with ATC do not specify "button-pressing" details of radio operation.

A *variable* in the work domain specifies the information needed as input to accomplish the work (input variables) or the states that can be directly affected as the result of the work (output variables). For example, input variables for setting a final approach course include ATC clearances and routes from a flight plan, while output variables include target settings for heading and autopilot. Input variables include those specifying the conditions in which a function applies (such as constraints) and those providing values for computation (such as current airspeed or last ATC clearance). Values of variables needed as input to the user to carry out a work function are typically provided in displays; values of output variables are typically set by an agent through controls, as the result of carrying out the function.

Individual work functions have associated input and output variables. A function's mapping from the input variables to the output variables is required to accomplish the function, but does not depend on variations in cockpit displays and controls. In simplest form, the association between work functions and variables can be represented as a

binary matrix in which rows represent work functions, columns represent input and output variables, and cells represent whether or not a particular variable is relevant to a particular work function. Each function (one row) flags the variables needed as input (columns for input variables), and the variables affected as the outcome of the function (columns for output variables). Similarly, cell values in one column indicate the functions (rows) that implicate that input or output variable. A matrix for cockpit aviation on a routine domestic flight has been built.

This representation prioritizes simplicity and ignores much information about the domain, including inter-function dependencies and sequence. It is intended to provide a necessary but not sufficient characterization of work needs, identifying, independently of a particular interface, what information and actions should be provided in displays and controls. The basic matrix could be extended to include weights for criticality, relevance or other variables.

Using the Work-Function X Variable representation to assess technology requires a representation of the technology that allows comparison to the work. The set of variables from the work representation provides vocabulary that can also be used to represent the technology being assessed. A device matrix analogous and comparable to the work function matrix can be constructed: rows are device components, columns are the variables, and cells indicate whether the row device-components expresses the column variable. Level of decomposition of the device, like level of decomposition of the work, is tailored to the evaluation goals. The device decomposition represents the technology in its functional components such as individual displays, controls, or decision aids. Device components might be defined as different spatial elements or elements available at different times.

Comparison between the Work Function Matrix and the Device Matrix can provide several types of information supporting validation.

Coverage: Are all the variables required by the work provided in the device? If not, what variables used by work functions are missing? (Are there column variables associated with a work function in the work matrix but not associated with any device component in the device matrix?) Coverage of the work may not be complete. Information might come from other sources (e.g., the user's observation) and effects might be accomplished by other means (e.g. another human). This may be acceptable or necessary in some situations, but broadly this identifies a problem or limitation in the device; alternatively it may motivate redefining the scope of work the device supports.

Overhead: Are all the input and output variables that are associated with device components actually relevant to the work? If not, what variables are specific only to the device? (Are there column variables associated with a device component in the device matrix but not associated with any



The level is at a higher abstraction level than evaluation object, and higher than specific procedures which would use that object to conduct work. It is intended to specify necessary but not sufficient needs of the target-work area and requires an abstract level of representation.

### **Specific Level Representation: Device Operation Example**

Highly specific representations of work are useful for evaluating a particular combination of device and operating procedures. Some work-areas are highly proceduralized, for example, specifying how equipment should be operated even beyond the physical requirements. While excessive proceduralization has been criticized, there can be valid operational reasons for procedures and required procedure adherence. Certainly there are practical demands for systems to support procedure execution. PRL is a procedure representation language designed to support mixed initiative automation, and it has been applied to procedures for operating equipment on the International Space Station (Schreckenghost et al, 2008). This provides a highly specific, procedure-based work representation and provides the foundation for assessing whether an implemented system is sufficient for effectively executing procedures. This can be done by collecting data about time and errors carrying out procedures (Billman, Schreckengost, & Miri, submitted). Validation of sufficiency could also be based on modeling human interaction, to determine whether it is possible to specify action sequences through an interface to accomplish a set of target procedures.

### **Conclusions**

Understanding variations in the characteristics of work domains and the purpose of analysis can guide the choice of work representations, and in turn the type of validation that can be done to assess whether the work expressed in that representation is adequately supported by the technology to be validated. Broadly, abstract representations typically focus on identifying necessary properties of work and on what a system must deliver to be validated. Specific representations typically focus on characterizing criteria sufficient to carry out the work and on what a system must deliver to be validated. Frequently mixed and multiple representations may be valuable. However, the most useful level of abstraction of work representations may be related to the distribution of routine versus generative work within a target-area.

### **References**

Billman, D., & Fan, C. (submitted). Displaying Work Structure: Biclusters of Work Functions by Work Variables.

Billman, D., He, H., & Owen, A. L. (submitted). Grouping pilot tasks and data display items via non-negative matrix factorization.

Billman, D., Schreckengost, D., & Miri, P. (submitted). *Alternative Interfaces for Commanding Spacecraft: Concept Assessment for Flexible Work Allocation*.

Bolton, M. L., & Bass, E. J. (2009). Enhanced operator function model: A generic human task behavior modeling language. In *2009 IEEE International Conference on Systems, Man and Cybernetics* (pp. 2904–2911). San Antonio, TX, USA.

Card, S. K., Moran, T. P., & Newell, A. (1983). *The psychology of human-computer interaction*. Hillsdale, N.J.: L. Erlbaum.

Charette, R. N. (2005). Why software fails [software failure]. *IEEE Spectrum*, 42(9), 42–49.

Ellis, S. R. (2000). Collision in Space. *Ergonomics in Design: The Quarterly of Human Factors Applications*, 8(1), 4–9.

Fennell, K., Sherry, L., Roberts, Jr., R. J., & Feary, M. (2006). Difficult Access: The Impact of Recall Steps on Flight Management System Errors. *The International Journal of Aviation Psychology*, 16(2), 175–196.

Kieras, D. (1997). Task analysis and the design of functionality. In A. Tucker (Ed.), *The Computer Science and Engineering Handbook* (Vol. 23, pp. 1401–1423). Boca Raton: CRC Inc.

Kirwan, B., & Ainsworth, L. K. (1992). *A Guide to task analysis*. London ; Washington, DC: Taylor & Francis.

Naikar, N., Hopcroft, R., & Moylan, A. (2005). *Work Domain Analysis: Theoretical Concepts and Methodology*. Melbourne, Australia: Australian Defence Science and Technology Org.

Palanque, P., & Bastide, R. (1997). Synergistic modelling of tasks, users and systems using formal specification techniques. *Interacting with Computers*, 9(2), 129–153.

Pew, R. W., & Mavor, A. S. (Eds.). (2007). *Human-System Integration in the System Development Process*. Washington, D.C.: National Research Council, The National Academies Press.

Rasmussen, J., Pejtersen, A. M., & Goodstein, L. P. (1994). *Cognitive systems engineering*. New York: Wiley.

Vallacher, R. R., & Wegner, D. M. (1987). What do people think they're doing? Action identification and human behavior. *Psychological Review*, 94(1), 3–15.

Schreckenghost, D., R. P. Bonasso, D. Kortenkamp, S. Bell, T. Milam, C. Thronesbery. (2008) Adjustable Autonomy with NASA Procedures. 9th International Symposium on Artificial Intelligence, Robotics & Automation in Space, Pasadena, h 2008

Vicente, K. J. (1999). *Cognitive work analysis : toward safe, productive, and healthy computer-based work*. Mahwah, N.J.: Lawrence Erlbaum Associates.

### **Acknowledgments**

This research was supported by funding through NASA Human Research Program Space Human Factors Engineering, Feary PI, Billman (Co-PI) and by the National Space Biomedical Research Institute through NASA NCC 9-58. Thanks to Chengcheng Fan, Michael Feary, Hera He, Art Owen, and Michael Stewart.