



# Alignment of Technology to Work: Design & Evaluation Representation

Dorrit Billman<sup>1</sup>, John Archdeacon<sup>2</sup>, Rohit Deshmukh<sup>1</sup>, Michael Feary<sup>3</sup>,  
Jon Holbrook<sup>1</sup>, Michael Stewart<sup>1</sup>

<sup>1</sup>San Jose State University, <sup>2</sup> Dell, <sup>3</sup>NASA Ames Research Center  
San Jose State University @NASA Ames, Moffett Field, CA 94035-1000  
[dorrit.billman@nasa.gov](mailto:dorrit.billman@nasa.gov)

**Abstract.** Robust and effective technology for high-stakes work depends on an appropriate understanding of the work needs. We summarize a method for representing work, technology, and their alignment, which aims to guide technology design and evaluation and to ensure that technology is fit-for-purpose. We illustrate with aerospace examples.

## 1 Introduction

Effective integration of humans and technology in safety critical systems requires an understanding of what functions the system is intended to support. Work needs should drive a) design, for example by driving requirement specification, and b) evaluation, for example through validation. When analysis of work needs guides design and evaluation, this should ensure effective, robust technology for carrying out the functions needed across a work domain.

In the domain of home entertainment, one intended function is recording for later viewing an upcoming event, such as the Singapore Grand Prix. Some technology designs may require the user to seek elsewhere information about time and local broadcast source, then specify time and channel on VCR, command the recording, and “manage the automation” by checking space limitation or other possible conflicts. A device better fit-for-purpose would provide all the information and operations for each function, accessible together. Issues of fitness-for-purpose can arise in any activity domain but are particularly important for high stakes and safety critical work.

The functions that the system should support might only emerge over the course of development and evaluation. It would be efficient, however, if the analysis preceded design and development, so missing or misappropriated functions are not encountered only in evaluation. Identifying and supporting work needs early in the design process prevents expensive redesigns and expensive training to compensate for a design not fit-for-purpose. A method

for identifying work needs early and a representation that can be shared and used throughout the design-to-delivery phases is very desirable.

This paper describes a method and representation of work, of technology, and their alignment, which we are currently applying and evaluating. The representation and method aim:

- to guide development of requirement specifications (or analogous documents) particularly in high-stakes work domains;
- to guide design; and
- to provide for ongoing evaluation, by assessing alignment of technology designs with the work.

Further, providing a representation of the work to be done provides input to the function allocation policy determining what agent does what components of work. When requirement specification is guided by work needs analysis, the requirements can provide a stable, shared representation throughout the specify-design-evaluate process, e.g. as in [1], that keeps the process anchored to actual needs of the work. Multiple researchers have noted the value of work representations and we have been influenced by many of these [2]–[7].

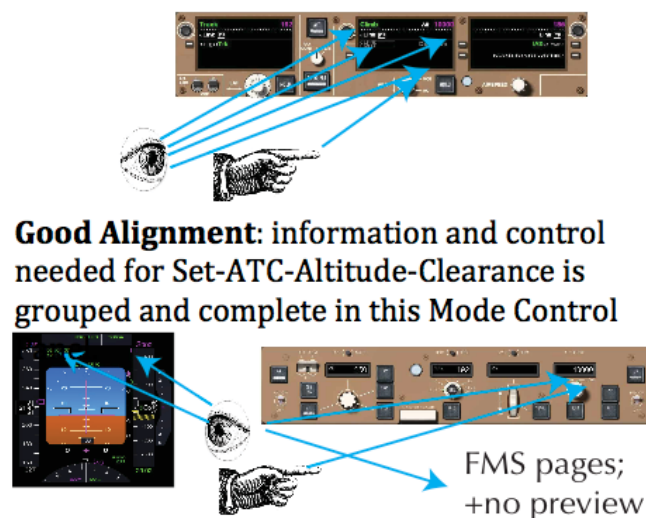


Figure 1. Illustrative example of good and poor alignment of cockpit elements use for tactical change in course, typically compliance with ATC clearances.

We claim that fitness-for-purpose depends on the alignment of the structure of the technology with the structure of the work. By structure we mean the components and their organization, of the work and of the technology. By alignment we mean that the components of work map relatively directly onto the components of technology. In turn this facilitates transparency of the technology, so that workers can focus on the work they wish to accomplish rather than the needs of the technology they are using to get the work done. For example, if the work domain is piloting autoflight-capable jets, a cockpit

design that is well aligned with the work will be organized around flight rather than the internal workings of the autoflight system. Of course, management of the technology (the autoflight system) becomes part of the work (flying an airplane), but well-aligned technology adds minimal and subsidiary demands. Figure 1 illustrates how technologies might differ in alignment for a single work function -- complying with an Air Traffic Control (ATC) clearance to change altitude. This requires as input variables the present altitude, the current target altitude, the altitude specified by ATC, and the current autoflight mode; it changes as output the current target altitude and possibly the mode. Alternative Mode Control Panels, as sketched, may be better or more poorly aligned with this function.

A key insight in developing our representation was recognizing that the input and output variables relevant to work functions can act as a “common language” to describe both the components of work and the components of the target technology. This common language mediates the assessment of alignment of technology to the work. In overview:

- The work and the technology are each represented as a matrix.
- The work matrix has the intended work functions listed as one dimension and the variables used or affected by the work on the other dimension.
- The technology matrix has the technology components listed as one dimension and the variables that the technology provides through displays and controls as the second dimension.
- Alignment can be assessed by comparing the components and structure of the technology matrix to the work matrix.

## 2 Work Functions, Variables, and Matrix

We represent work in terms of a set of intended *work functions* and a set of *variables*, within a bounded *work domain*. A *work domain* is a body of related activities intended to accomplish a set of stable goals, such as making products or controlling processes.

An intended *work function* is an activity needed to accomplish the mission or achieve the goals of a work domain. To be the most useful guides for technology design and evaluation, work “functions” are more abstract than what are typically referred to as “tasks.” New technology can transform work, and work functions should be written at a level sufficiently abstract to allow variation in how the same function is to be accomplished across alternative technology designs. Granularity of the work functions is an important analysis choice, influenced by the analysis purpose and the scope of the technology;

more specific breakouts of functions might be undertaken within a larger analysis as well.

Work functions are represented in terms of the variables needed as input to and the variables affected as output from the function. Input variables represent the information and resources needed to do the work. For cognitive work, information variables are typically the key concern in designing and evaluating technology, that is, much of the design challenge concerns how to make available all the information needed to make decisions without producing information overload. Output variables are changed as the result of carrying out a work function. Output variables broadly map onto the aspects that can be affected by any control action. Output variables can change the state of the interface, the state of the system behind the interface, an information variable such as a decision, or the information made available to different agents, as through calculation or communication. Identification of the *input* variables needed across a work domain has also been stressed by developers of ecological interface design, as the *information requirements* [8]. We also recommend a parallel census for output variables. In the aviation case we have looked at in most detail, the number of input variables seems to be substantially higher than the number of output variables.

Given a set of work functions and a set of the input and output variables, each work function is coded for what information variables it needs for input and what output variables express the changes accomplished by the function. Consider the function of executing a highly planned attitude manoeuvre of the International Space Station (ISS) to accommodate a Soyuz docking. Input variables would include the systems to be monitored to ensure that the planned and actual situations align: S-band communication availability, status of the momentum management system on the ISS, status of controlling software on the ISS, trajectory of the approaching Soyuz, status of the software controlling Soyuz, status of crew, and status of interrelated systems on the ISS such as configuration of the solar arrays. Output variables would include issuing commands to the momentum management system and providing information to ground controllers about progress. Some variables may be complex and derived as the output of other work functions, such as a summary representation that a particular system is 'normal'. Our concept of work functions with input and output values is quite similar to Pritchett et al's [9] representation of work functions.

Given a census of work functions and of the input and output variables for a work domain, the relation between these can be expressed in a matrix. A work function can be represented as a vector of variables, with 1s indicating a variable is relevant to the function, and 0s otherwise. Figure 2-A illustrates

the individual work function vectors integrated into a matrix. The top line in the matrix represents work function 1 as the row vector (0,0,0,1,0,0,0,1,1,0,0,0,1). Note this vector does not specify *what* the value of the variable is, but simply that the variable is important for carrying out that function. Conversely, a variable can be represented as a column vector indicating what work functions use that particular variable, the vector (0,0,0,0,0,0,0,0,1,1) for input variable 1.

**Work domain variables**  
info needed      changes effected

| Work domain functions | Work domain variables |             |             |             |             |             |             |             |              |              |              |              |              |
|-----------------------|-----------------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|--------------|--------------|--------------|--------------|--------------|
|                       | input var 1           | input var 2 | input var 3 | input var 4 | input var 5 | input var 6 | input var 7 | input var 8 | output var 1 | output var 2 | output var 3 | output var 4 | output var 5 |
| work function 1       |                       |             |             | 1           |             |             |             |             | 1            |              |              |              | 1            |
| work function 2       |                       |             | 1           |             |             | 1           |             |             |              |              |              | 1            |              |
| work function 3       |                       |             | 1           |             |             |             | 1           |             |              |              |              | 1            |              |
| work function 4       |                       |             |             | 1           |             | 1           | 1           |             |              |              |              | 1            |              |
| work function 5       |                       |             |             |             |             |             |             |             |              |              |              |              | 1            |
| work function 6       |                       |             | 1           |             |             |             |             |             | 1            |              |              |              | 1            |
| work function 7       |                       |             |             |             | 1           |             |             |             | 1            |              |              |              |              |
| work function 8       |                       |             |             |             |             | 1           | 1           |             |              |              |              |              |              |
| work function 9       | 1                     |             |             | 1           |             |             |             |             |              | 1            |              |              |              |
| work function 10      | 1                     |             |             |             |             |             |             |             | 1            |              |              |              | 1            |

Figure 2A: Work Matrix

**Variables use to represent technology**  
info provided      changes effected  
("Displays")      ("Controls")

| Device Components | Variables use to represent technology |             |             |             |             |             |             |             |              |              |              |              |              |
|-------------------|---------------------------------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|--------------|--------------|--------------|--------------|--------------|
|                   | input var 1                           | input var 2 | input var 3 | input var 4 | input var 5 | input var 6 | input var 7 | input var 8 | output var 1 | output var 2 | output var 3 | output var 4 | output var 5 |
| component 1       | 1                                     | 1           | 1           |             |             |             |             |             |              |              | 1            |              |              |
| component 2       |                                       |             |             | 1           | 1           |             |             |             |              |              |              | 1            | 1            |
| component 3       |                                       |             |             |             |             | 1           | 1           | 1           |              |              |              |              | 1            |
| component 4       |                                       |             |             |             |             |             |             | 1           | 1            |              |              |              | 1            |

Figure 2B: Technology Matrix

### 3 Technology Components

Just as a work domain can be decomposed into component work functions, a complex technology can be decomposed (or built up) from its components. A component is a group of interaction elements, i.e., a set of displays and controls, related by spatial or temporal proximity. Both spatial (e.g., located close together or within the same physical unit) and temporal (e.g., information appearing on a screen at the same time) proximity of elements are important in interaction [5] and in setting component boundaries. Temporal proximity and distance is relevant when the interaction options change with time, enforcing sequential steps to access; for example, requiring navigation through multiple screens to access one from the other or if mode changes restrict what resources are available at one time. Spatial proximity is more studied in automation [10], and it would be unusual to group together as one component, elements that are widely separate spatially. We believe elements of a complex automation system are often marked jointly by spatial proximity, simultaneous availability, and development or manufacturing history. However, as with identifying the domain work functions and variables, picking the size, or level, of the components requires judgment and will depend on the purpose of the analysis. For design, identifying higher-level

parts initially, and adding a detailed analysis of selected components may be helpful. For evaluation, definition of lower level components may be helpful. For simplicity we assume that components at the within one analysis are disjoint.

The same variable set is used for representing technology components as for representing work functions, providing a common language. Just as a work function can be scored for which variables are relevant, a technology component can be scored for the displayed information (providing input to the user) and controllable variables (enabling action by the user) it provides. Note that the technology *displays* provide work function input-variables and the technology *controls* provide work function output-variables. As with the vectors for individual work functions, an individual technology component can be scored as a vector of 1's and 0's showing the variables it supports. Again, this vector does not specify *what* the value of the variable is that the component provides, but simply that the variable is provided by the component. Conversely, a variable can be represented as a vector indicating what components provide support for it. Typically, components also include displays and controls that do not have to do with the work domain but only with operation of the technology. These can be thought of as “overhead”; typically the variables used to represent the components will be the union of the domain and the overhead variables.

Just as the work domain can be represented as a matrix, so can the technology. The matrix representation of work and technology, with shared dimension of variable, supports analysis of alignment.

## 4 Alignment of Technology with the Work Domain

There are three aspects of alignment: coverage, overhead, and organization. **Coverage** refers to the proportion of work domain variables that are provided somewhere in the technology. Technology with low coverage cannot go very far supporting the breadth of work functions. While it may not always be desirable or possible for technology to provide maximal coverage (some variables may be better left to a person), it is valuable in any case to have an explicit accounting of what variables the technology is and is not supporting. Technology with higher coverage is better aligned and typically better able to support its intended purpose, *ceteris paribus*.

**Overhead** refers to the variables that are not concerned with getting domain work done, but exclusively concern operation of the technology that is the focus of the design or evaluation. The definition of overhead critically depends on the scope of the work domain. If the work domain is work of the cockpit

crew piloting an autoflight-capable jetliner, the work functions describe what is needed to fly the jetliner on the intended trajectory with appropriate communication; the overhead measures the work needed to manage the devices in the cockpit such as switching among modes or accessing information in the flight plan. Technology with high overhead may be “rude” or awkward to use. Technology with lower overhead is better aligned, and typically better able to support its intended purpose.

**Organization** refers to the way that the work functions and the device components group and are grouped by the variables. In a simple domain, there might be a one-to-one mapping between work functions and technology components. Every work function would have its own technology component, providing support for all the relevant variables, whether a separate web page or a separate, specialized tool for a physical assembly job.

In a slightly more complicated situation, several work functions might be accomplished with the same, single component, producing a many-to-one mapping. For complex work the “best” design may provide a many-to-many mapping: there may simply be too many combinations of variables needed by different work functions to make a one-to-one mapping feasible. However, mapping clusters of related variables onto a component can still produce a well-aligned organization: the variables grouped together in one component are organized to provide most of the variables needed by most of the functions in that cluster. If there is little similarity structure across work functions—each picking a very different set of variables from the others, there is a very limited possibility for aligned organization. The technology design could pick a few high-stakes or frequent work functions, design to make support for these few coherent (for however much complexity can be tolerated) and leave the remaining functions to “piece together” their needed variables, which are scattered across components.

Some aspects of organization can be “read off” from a matrix through sorting and inspection. Disjoint sub-matrices may reveal clearly separable groups of functions, which share variables within but not between function groups. However, recovering the implicit matrix organization is difficult to do by inspection and we have been developing and applying (bi) clustering methods, focusing on discovery of structure within the work function matrix.

A preliminary analysis of alignment was developed for aspects of the aviation domain, particularly for an airline pilot’s tactical changes in trajectory in response to Air Traffic Control clearances.

## 5 Exploratory Evaluation: Effect of Alignment on Ease of Learning

Rationale. Technology that is better aligned with a work domain (or part thereof) should be easier to learn for domain experts. That is, the more that working *through* the technology is based on work activities (already familiar to domain experts), the less that operation requires learning new, technology-specific skills and knowledge. One aspect of alignment is whether, work function by work function, the technology provides displays for the information variables and controls for the output variables for each work function; and whether the interface groups together those elements that are needed together.

Method. We identified and evolved two designs for Mode Control Panels (MCPs). MCP's are intended for use by an airline pilot to enter tactical changes in the airplane's trajectory, typically in response to Air Traffic Control clearances. These two MCP designs differed in how directly they were aligned with the needs of work functions for complying with ATC clearances. One important difference between the two MCP designs was the organization of displays and controls in the interface. In the cockpit design with the *Alpha* MCP the displays and controls for work functions such as compliance moving to a newly specified altitude were distributed across several cockpit devices (the MCP, the Primary Flight Display, the Flight Management System); in the cockpit design with the *Bravo* MCP the displays and controls for such ATC clearances were integrated within the MCP, hence spatially close and available without paging or navigating. Thus, we anticipated that learning the particular actions needed to implement ATC clearances would be faster in a cockpit using the Bravo rather than Alpha MCP, for 1) pilots needed to implement ATC clearances would be faster in a cockpit using the Bravo rather than Alpha MCP, for 1) pilots familiar with ATC clearances and flying in the National Air Space but 2) unfamiliar with either MCP design.

We developed 8 scenarios, flown in a mid-fidelity simulator (Figure 3). Six involved vertical and speed changes (where we anticipated most difference) and 2 involving lateral changes. We used a trials-to-criterion learning procedure with a criterion of 2 successive correct executions of a particular scenario. The scenarios were repeated, in sequence, with individual scenarios dropping out as the participant reached criterion on that scenario. Users were corrected when they made an error and the scenario stopped. Participants were 6 regional pilots who had experience with autopilot aircraft, but were screened to be unfamiliar with the Alpha interface (similar to one in commercial use).

Follow-up questioning found that User #3 (in Figure 4) unfortunately did in fact have experience relevant to the Alpha interface.



Figure 3. Current simulator cab with lab-designed Prototype Design. Participant sits at left (Captain’s seat) and experimenter at right (First Officer seat). Experimenter uses tablet computer to control progress through the experiment.

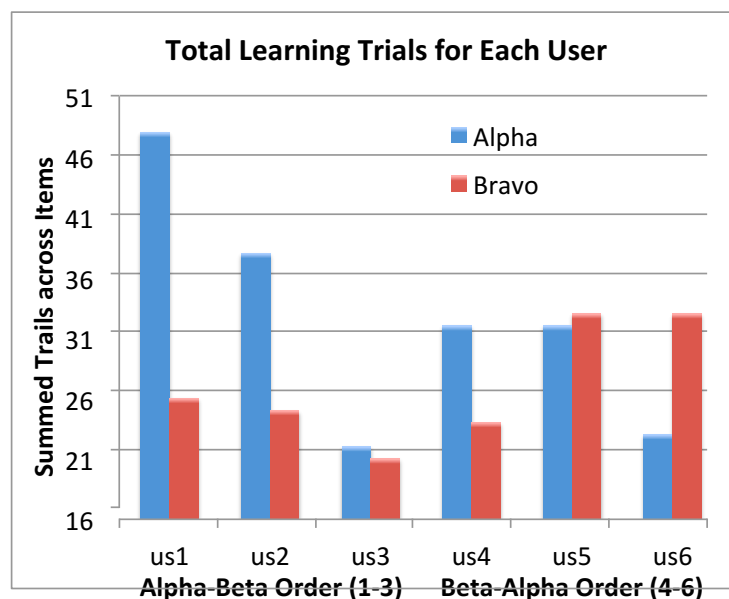


Figure 4. Trials-to-criterion with Alpha and Bravo MCP designs. Users are grouped by which MCP was used first. Users 1-3 used Traditional first; users 4-6 used Prototype first. The figure suggests better learning on the second interface and better learning for Alpha, substantiated in analyses.

Results and Analysis. Figure 4 shows trials to criterion for cockpits with each MCP design, grouped by which system the participant learned first. We modelled the factors influencing trials-to-criterion with linear mixed model regression, which offers a sensitive analysis for within-subject and small N designs. We used R (R Core Team, 2012) and *lme4* [11], with the

untransformed trials-to-criterion as the variable predicted. Fixed effects in the model were MCP Condition (Traditional vs Prototype), Order (First vs Second experience), and Item. Subject (intercept only) was a random effect. Using the *Bravo* MCP design reduced the learning trials by 0.71 trials (StDev = 0.2716, associated  $t = -2.608$ ). To test the significance of the effect of Condition, we used the maximum likelihood approach comparing this model to the null model from which Condition had been dropped, and found that Condition is a significant factor ( $\chi^2(1)=6.557$ ,  $p=0.01045$ ). An analogous analysis showed that Order was also significant ( $\chi^2(1)= 8.9537$ ,  $p=0.002769$ ); users were faster learning the second system they were introduced to, with this reducing the trials to criterion by 0.833 trials (StDev=.2716,  $t=-3.068$ ).

This preliminary, low  $n$  study provided suggestive evidence for ease of learning of the Bravo over Alpha design, using a trials-to-criterion learning measure. Limitations include small  $n$ , few scenarios, use of one learning method, and imprecision in scope of the work functions assessed. Nevertheless, this study suggests that the alignment of the technology's content and organization to the work domain's content and organization can be assessed and that alignment may be important for learnability.

## 6 Conclusions & Future Research Needs

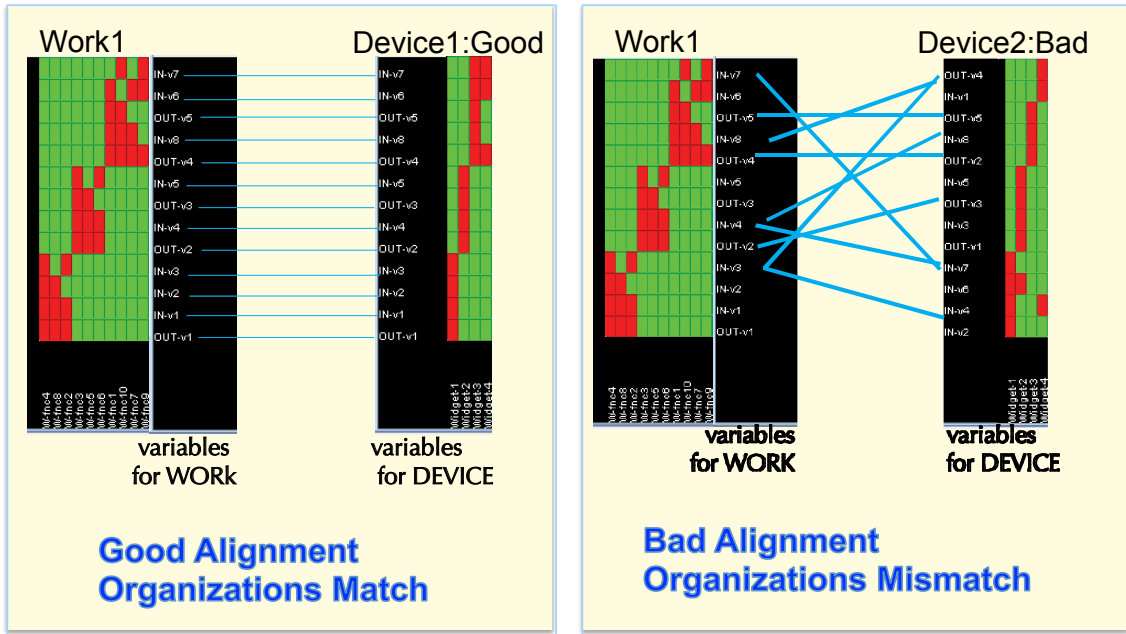
To ensure resilience of safety-critical systems, the integrated performance of the human-plus-engineered components must reliably support the functions needed in the work domain. This requires understanding the work, the technology, and how the technology is related to the work. Each of these is complex, and there are complementary perspectives for addressing complexity: depth first or breadth first. A 'depth first' perspective focuses on a sample of key elements in the work and in the technology. Each of the sampled scenarios may be evaluated in detail, but it may be infeasible to assess the whole work domain in such depth. A 'breadth first' perspective focuses on covering all aspects of a well-defined work domain and candidate technology, though in less detail. In our 'breadth first' approach, we use a relatively simple representation of both work and technology. A "common language," for the informational variables needed to make decisions and the action variables needed to effect changes, can describe and compare the work functions and the technology support at a relatively abstract level. An important research goal will be exploring these complementary perspectives and, particularly, how they can be usefully coordinated.

## Acknowledgements

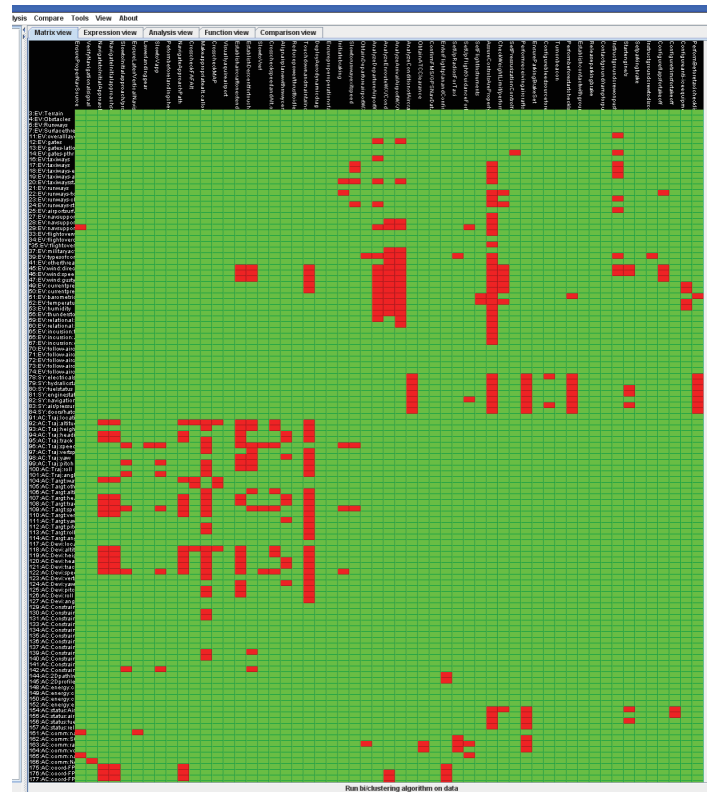
This research was funded in part by the NASA Human Research Program: Space Human Factors Engineering (466199.02.01) PI, Michael Feary. Thanks to members of the AID lab group. Thanks to Emilie Roth for comments on an earlier draft.

## References

- [1] R. W. Pew and A. S. Mavor, Eds., *Human-System Integration in the System Development Process*. Washington, D.C.: National Research Council, The National Academies Press, 2007.
- [2] K. A. Butler, J. Zhang, C. Esposito, A. Bahrami, R. Hebron, and D. Kieras, “Work-centered design: a case study of a mixed-initiative scheduler,” in *Proceedings of the SIGCHI conference on Human factors in computing systems*, 2007, pp. 747–756.
- [3] K. Holtzblatt, J. B. Wendell, and S. Wood, *Rapid contextual design: a how-to guide to key techniques for user-centered design*. San Francisco: Elsevier/Morgan Kaufmann, 2005.
- [4] H. Johnson and P. Johnson, “Task knowledge structures: Psychological basis and integration into system design,” *Acta Psychol. (Amst.)*, vol. 78, pp. 3–26, 1991.
- [5] D. Kieras, “Task analysis and the design of functionality,” in *The Computer Science and Engineering Handbook*, vol. 23, A. Tucker, Ed. Boca Raton: CRC Inc., 1997, pp. 1401–1423.
- [6] N. Naikar, R. Hopcroft, and A. Moylan, “Work Domain Analysis: Theoretical Concepts and Methodology,” Australian Defence Science and Technology Organisation, Melbourne, Australia, 2005.
- [7] E. M. Roth and R. G. Eggleston, “Forging New Evaluation Paradigms: Beyond Statistical Generalization,” in *Macro-cognition Metrics and Scenarios Design and Evaluation for Real-World Teams*, E. S. Patterson and J. E. Miller, Eds. Surrey, England: Ashgate Publishing, 2013.
- [8] C. M. Burns and J. R. Hajdukiewicz, *Ecological interface design*. Boca Raton, FL: CRC Press, 2004.
- [9] A. R. Pritchett, S. Y. Kim, and K. M. Feigh, “Measuring Human-Automation Function Allocation,” *J. Cogn. Eng. Decis. Mak.*, vol. 8, no. 1, pp. 52–77, Mar. 2014.
- [10] C. D. Wickens and A. D. Andre, “Proximity Compatibility and Information Display: Effects of Color, Space, and Objectness on Information Integration,” *Hum. Factors J. Hum. Factors Ergon. Soc.*, vol. 32, no. 1, pp. 61–77, Feb. 1990.
- [11] J. Pinheiro, D. Bates, S. DebRoy, and D. Sarkar, *Fit and compare Gaussian linear and nonlinear mixed-effects models*. R Package Version nlme 3.1-118, 2014.



APPENDIX 1 *Alignment Illustration*. The left panel shows a technology perfectly aligned with the work functions. The technology groups displays and controls for these variables to match the grouping based on work functions. The right panel shows bad alignment. Some variables in the work domain are not covered, the device includes overhead variables, and the way the displays & controls group variables is unrelated to the way the work functions group variables.



APPENDIX 2. Part of the 119 x 210 function X variable matrix for cockpit aviation. Inspection suggests loose cluster structure, where sets of work functions draw on sets of common variables. The visual patterns suggest the relevance of statistical clustering methods to identify domain structure .