



LUND UNIVERSITY

Applying Phenomenology and Hermeneutics in IS Design: A Report on Field Experiences

Whitaker, Randall

Published in:

Use and Redesign in IS: Double Helix Relationships?

2007

[Link to publication](#)

Citation for published version (APA):

Whitaker, R. (2007). Applying Phenomenology and Hermeneutics in IS Design: A Report on Field Experiences. In H.-E. Nissen, P. Bednar, & C. Welch (Eds.), *Use and Redesign in IS: Double Helix Relationships?* (pp. 63-96). Informing Science Press.

Total number of authors:

1

General rights

Unless other specific re-use rights are stated the following general rights apply:

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

Read more about Creative commons licenses: <https://creativecommons.org/licenses/>

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

LUND UNIVERSITY

PO Box 117
221 00 Lund
+46 46-222 00 00

Applying Phenomenology and Hermeneutics in IS Design: A Report on Field Experiences

Randall Whitaker
EnolaGaia.com, Dayton, Ohio, USA

EnolaGaia@aol.com

Abstract

Phenomenology and hermeneutics have long been promoted as sources of inspiration for better information system (IS) design. Practical approaches to applying these philosophical ideas have been awaited for just as long. This paper offers a review of one practitioner's experience in devising a theoretical and methodological 'toolkit' via which these disciplines' principles have been applied in actual IS design practice.

Keywords: design, phenomenology, hermeneutics, information systems, praxio-focal approach.

Introduction

One December night circa 1990, on a night train from Stockholm to Uppsala, I struck up a conversation with a German scholar returning from that evening's Nobel Prize ceremony. By way of introduction, I told him I was a researcher whose interests focused on phenomenological approaches to epistemology and their application to the design and use of information systems. He was quite familiar with such thinkers as Husserl and Heidegger, and he expressed surprise that computer scientists would know of them - much less seek to apply their ideas.

Material published as part of this publication, either on-line or in print, is copyrighted by the Informing Science Institute. Permission to make digital or paper copy of part or all of these works for personal or classroom use is granted without fee provided that the copies are not made or distributed for profit or commercial advantage AND that copies 1) bear this notice in full and 2) give the full citation on the first page. It is permissible to abstract these works so long as credit is given. To copy in all other cases or to republish or to post on a server or to redistribute to lists requires specific permission and payment of a fee. Contact Publisher@InformingScience.org to request redistribution permission.

When I inquired about his academic specialty, he said it was a very obscure field probably unknown to me - hermeneutics. He described it as an esoteric discipline he enjoyed, even though it rendered him something of an isolated party even within academe. He was stunned to hear I had been reading Gadamer and that other information technology researchers were promoting hermeneutics as a source of inspiration and guidance in addressing how IT artifacts are interpreted via design and in practice. By the time we had parted company, he was visibly energized by the idea his little-known discipline might enjoy a renaissance through application to IT.

Since that evening, two things have happened. First, I have gone on to a career as a senior human factors / cognitive engineering researcher with a major American defense contractor. Over the last several years I have been involved in a progressive series of projects which have afforded me the opportunity to freely employ my theoretical inclinations (phenomenology, and to a lesser extent hermeneutics) in actual IS development practice. In the course of these projects I have had to confront obstacles and exigencies in both (a) applying these theoretical orientations to 'real-world' situations and (b) demonstrating their unique contributions to the projects' consistently successful outcomes.

Second, there has been ongoing promotion of phenomenology and hermeneutics in the literature of such relevant fields as human-computer interaction (HCI), design science, and participatory design (PD). Though voluminous, this literature can be characterized as 'scholarly' - i.e., abstract or theoretical in content. Fine points of philosophy have been examined in relative isolation from consideration of how they might pertain to the workaday world. Regardless of its illumination of 'meaning' or 'reflection', such theoretical work rarely addresses either the 'meaningful use' or 'reflection upon use' at the center of both my professional design work and this monograph's theme. As a result, such scholarly work - meritorious though it may be - has provided little aid in applying phenomenology and hermeneutics to IS design and in justifying their relevance to the orthodox IS development community.

This monograph's call for contributions cited the "dialectic between meaningful use and reflection upon use." My purpose in this paper is to address that dialectic with regard to my years of attempting to apply phenomenology and hermeneutics in IS design. In this case, 'meaningful use' refers to application of those approaches in IS design and de-

velopment, and 'reflection upon use' refers to what I've derived from those experiences - both in terms of methodological development and feedback on demonstrable applicability of phenomenological and hermeneutic principles. I shall attempt to illustrate these dual sides of the 'double helix' with some selected topics from my practical experience. The objective is not to 'prove' the utility of phenomenology and hermeneutics per se, but rather to report on what I've learned about these orientations' demonstrable 'utility'.

Graduating from Theory to Praxis

The most basic problems I've encountered in applying phenomenological and hermeneutic concepts to IS development relate to the distinction between 'theory' and 'praxis'. Both these fields emphasize subjectivity - a topic adequately addressable and typically addressed in the abstract, with little regard to everyday action. In other words, the literature offers much in the way of elaborating 'theory' but little in the way of informing 'praxis'. This emphasis on 'theory' over 'praxis' is understandable, because work on these ideas originated and largely remains within the scholarly realm. If I were still in academe, this state of affairs would cause me no concern. However, I am a scholarly practitioner rather than a practicing scholar, and this obligates me to confront questions of whether, and to what extent, these ideas can inform or even improve my working praxis.

The Problem of 'Grounding' Philosophical Theory to Support Praxis

Both phenomenology and hermeneutics focus on the subjectivity of a given person or actor. Phrased another way, these orientations frame their inquiry with regard to someone's 'first person perspective' (1PP, to borrow a term from game programmers). There are a variety of thinkers and theories connoted by the term 'phenomenology', and there is a corresponding diversity in their approaches to addressing this 1PP. The works of both Husserl and Heidegger illustrate the extreme of abstraction in their treatments of what can be present to experience (Husserl) and the actor's essential mode of being (Heidegger). Insightful though they are in educating IS professionals on human cognition and activity, neither offers much that can be directly incorporated into practice. Similarly, hermeneutics elucidates interpretation and inter-

pretability without providing specific tools or methods for leveraging the understanding one can obtain from its study.

In large part, this apparent deficiency derives from these fields' foci. Both phenomenology and hermeneutics emphasize processes and elements intrinsic to the given actor - e.g., what she perceives, the manner in which she thinks, or her capacity for interpretation. This imposes a methodological problem, because an analyst cannot directly inspect the target user's mind or thoughts. The analyst must address the target actor from a third-person perspective (3PP) from which any characterization of that actor's 1PP or 'phenomenology' is at best an allusion and at worst an illusion.

This does not mean philosophy cannot offer practitioners inspiration or guidance. One notably illustrative counterexample is Heidegger's concept of 'breakdown', in which the flow of a subject's phenomenal activity is perturbed out of a relatively automatic mode into a more consciously deliberative mode. Figuratively, such breakdown events have been recommended as observable symptoms of phenomenological perturbation and clues to problematical situations in IT usage (e.g., Ehn, 1988; Winograd & Flores, 1986). Such breakdown events can be observed as (e.g.) interruptions or digressions in the course of a given activity, and marking their occurrence is an excellent heuristic for identifying issues to be addressed and resolved.

The scope of philosophical concepts that can be brought to bear must therefore be qualified with respect to what can be evident to observation or reasonable inference. The example of breakdowns illustrates one type of such evidence - an impact on observable behavior in the course of an activity. This 'anchors' or 'grounds' the theoretical construct by correlating it with something empirically discernible. Once such 'grounding' can be achieved, IS professionals can pursue development of methods and tools geared to the type of evidence thus nominated for collection. Unfortunately, there are few examples of constructs or concepts from philosophical treatments of phenomenology and hermeneutics that are capable of such 'grounding'. The main problem is that the majority of such constructs are difficult to correlate with observable evidence. The secondary problem is that many elements of phenomenological or hermeneutic philosophy are offered as tenets whose acceptability is based on logical coherence or expository force, not testability against real world situations. As such, guidance for

interrelating explanatory abstractions to actual circumstances is rarely provided.

Achieving 'Grounding' via Wider Theoretical Foundations

The first problem in applying phenomenology and / or hermeneutics in IS design is therefore the identification of theories both (a) consistent with the tenets of phenomenology and hermeneutics and (b) offering insights or models that can be applied to guide praxis. My own approach has been to seek such theories regardless of whether they are popularly categorized as specimens of phenomenological or hermeneutic philosophy. The ones that have come to comprise my theoretical toolkit have recommended themselves on the basis of both (a) accommodating a focus on the subject actor's 1PP while immersed in the target activity and (b) providing a basis for correlating phenomenological elements with some basis for empirical observation.

With respect to theoretical foundations, my own preferred sources are drawn primarily from second-order cybernetics. They include the *cybernetics of cybernetics* of Heinz von Foerster (1981), the *radical constructivism* of Ernst von Glasersfeld (1995), and most particularly the *biology of cognition* and *enactive cognitive science* of Humberto Maturana and Francisco Varela (Maturana & Varela, 1980; Varela, 1979; Varela, Thompson, & Rosch, 1991). All address the 'phenomenology' and epistemological processes associated with a given system with respect to that system's 1PP. In particular, the work conducted by Maturana and Varela (both jointly and individually) offers a rich set of coherent constructs for characterizing the 'phenomenology of the living' - i.e., experience as contextualized with regard to an observing organism's biological constitution.

Another source of theoretical inspiration is the work of Charles Sanders Peirce (e.g., 1935), the first scholar to label philosophical examination of essential mental elements 'phenomenology'. Peirce's work was integrated with his semiotic theories, and this connection affords his 'phenomenology' a stronger linkage to issues of signs and signification than one derives from other philosophical works. This linkage affords the basis for applying Peirce's theory of signs where the form and interrelationship of symbolic elements is critical - as is the case with any IS project.

Additional theoretical support is drawn from work on design and development practices rather than phenomenology and hermeneutics per se. These influences have proven useful owing to their accommodation of - and even direct allowance for - the subjective aspects of a worker's experience. Over the years, the two I've found most useful are *participatory design (PD)* (Docherty, Fuchs-Kittowski, Kolm, & Mathiassen, 1987; Ehn, 1988; Greenbaum & Kyng, 1991) and *soft systems methodology (SSM)* (Checkland, 1981). Participatory design recommends itself for its emphasis on grounded work praxis and the worker's experience as a key object of study. Soft systems methodology recommends itself as a structured protocol actively seeking to portray the subject matter in the same terms as it is perceived and interpreted by workers.

These examples are sufficient to illustrate how anyone seeking to apply phenomenology and / or hermeneutics in IS design will need to consider relevant ideas from multiple disciplines. Some of the fields I've cited are at best peripheral to current IS design and development curricula, while others are so distant as to represent knowledge no student is likely to encounter at all. As a result, my own experience indicates *it requires considerable effort and creativity to equip oneself with a theoretical toolkit for phenomenologically or hermeneutically informed design.*

A Comparison of Perspectives and Methods

Adopting a more phenomenologically- or hermeneutically-oriented orientation entails changes or re-orientations with respect to methodology as well as theory. In this section I shall summarize some key points upon which I've learned a phenomenologically or hermeneutically informed orientation distinguishes itself from the prevailing approach to IS design and development. This will be done by way of a comparative review.

For the sake of illustration this review will be framed with regard to the prevailing model for IS design and development projects. Though documented in myriad specific formats, this model has a certain features that for all practical purposes may be considered universal. It consists of an essentially linear process path leading from initial study of the target use environment (*knowledge acquisition - KA*) through some form of *pro forma analysis* to the creation of a design concept and generation of design specifications. These specifications serve to guide software development and provide a baseline for testing and evaluating the eventual product (e.g., a prototype).

Methodological Tactics Reflective of Current Conventional Orientations

The linearity of the conventional process path insinuates its constituent activities must be conducted in a stepwise fashion and interconnected via a unidirectional feed-forward of results from one phase to the next. The overall process is therefore structured similarly to an assembly line - a pertinent metaphor, given that IS methodologies have been created and rendered orthodox by a population of professionals (e.g., engineers) self-defined as creators of concrete products.

This production metaphor is reflected in a focus on the product (an artifact). Knowledge acquisition and analysis are framed with regard to illuminating factors recommending the prospective artifact's features or configuration. Design is taken to be the process of prescribing such features or configurations, and development is taken to be their realization in the form of a new artifact. Regardless of the terms in which project motives were originally described, improvements to a work environment are in the end re-interpreted in terms of the product's novelty. This emphasis on novelty underlies the prevailing belief that an IS design and development project's prime objective is *innovation*.

The workers to whom these new artifacts are to be issued, if considered at all, are addressed as 'users' - i.e., in terms of their role as actors interacting with the artifact. To the very limited extent phenomenology or hermeneutics are ever invoked, it is in passing reference to the 'user experience' with, or 'usability' of, the artifact. The reduction of overall worker experience to only that portion involving the artifact both (a) eliminates concern for how the worker engages the data it provides and the work it supports while (b) simplifying evaluation by attending only to observable behaviors ascribed to human components of a subsuming work system. It is therefore fair to claim conventional IS methodological models are predicated on the metascientific orientation Radnitzky (1970) labels 'logical empiricism' - an objectivist orientation inimical to addressing concerns such as individual phenomenology or subjective interpretation (Whitaker, 1992).

This orientation influences the entire IS development process path. Of particular concern is the manner in which it affects the front-end knowledge acquisition activities. Subject matter experts (SME's) are treated as sources of objective data on the target work activity and the related functions to be supported with the new IS artifact(s). Freed of

concern for subjective aspects of work, analysts presume they can simplify the laborious KA phase through either or both of two tactics. The first is to rely on information sources other than the person(s) actually performing the target work activities. These sources may be (e.g.) other personnel more available for interviews or documentation of the 'official' version of the work and its conduct. The former may have little knowledge of current practices and may even provide views colored by their own (off-topic) experiences or speculation about actual praxis. Documentation is all too often out of date or uninformative about what individual workers have to engage and have to do to accomplish the target tasks.

The second simplification tactic is to conduct KA through interviews and other procedures that do not involve on-site observation of the target work being performed. Such approaches are usually justified on the basis of relative economy; direct observation, apprenticeship, and other immersive KA techniques are extremely time-consuming and laborious. Though these approaches can certainly capture valuable data in general, they stand little or no chance of uncovering context-specific aspects of the target work when conducted in isolation from the active work milieu.

The logical-empiricist orientation also fosters a reliance on formal models presumed capable of representing and / or illuminating the subject matter. The general notion of modeling the subject matter is not itself the problem. The problem lies in the fact that the most popular modeling approaches treat workers as 'black box' components within a work system or environment which is taken to be an appropriate and sufficient focus for a constructive model. A particularly relevant example is that of the *means-ends hierarchy* (or *abstraction hierarchy*) created by Jens Rasmussen (1986) and widely employed in cognitive systems engineering circles. This model lays out the interrelationships between a system's objectives and the elements involved in those objectives' accomplishment. The end result is a representation of a work system architecture from which individual actors have been eliminated as objects of reference. The abstraction hierarchy is a fine tool for what it is. However, it definitely is *not* a good tool for uncovering and analyzing the cognitive, experiential, or context-dependent aspects of work as engaged by the individual actor.

The net effect of these and other aspects of a logical-empirical approach is to progressively, if not comprehensively, emphasize the form

and function of a new artifact over the process and procedures of the work activities this artifact was commissioned to support. By the time a software prototype is produced it has become the fixed point of reference around which all issues revolve. This culminating position is well-illustrated by current evaluation practices centered on testing a product's 'usability' – i.e., the degree to which a 'user' can efficiently and effectively operate the IS product. For such 'usability' to be construable as definitive evidence of success, any concerns for the interrelationship between a person (in her work-demarcated role as a 'worker') and her task must have been supplanted by attention to the interrelationship between the person (in a distinct artifact-demarcated role of 'user') and the artifact itself. The risk inherent in this re-orientation is, of course, the deployment of artifacts certified as 'usable' yet demonstrably less than 'useful' in workaday practice.

Methodological Tactics Reflective of a Phenomenological Orientation

It is easy to characterize conventional IS practices as being geared to production of an artifact. It is more difficult to so concisely describe objectives from a phenomenological or hermeneutic orientation. One thing is certain - proceeding with this orientation requires demoting the eventual artifact from the status of a focal objective to that of an emergent outcome. Indeed, the very necessity for a new artifact must remain an open issue to be decided in light of what one learns of the worker's needs as seen from the worker's perspective. Phrased another way, the fact and the form of the artifact must be treated as a 'variable' whose instantiation is subordinate to the subjective experience of the worker in the course of her work. It is this flow of interwoven cognition and action which must be addressed as the fixed point of reference if IS designers are to have any chance of understanding current praxis and identifying clues to constructive change.

Knowledge acquisition and analysis must be framed with regard to illuminating how the work subject matter is apprehended, how the course of the work process unfolds, and how the process path is navigated from the vantage of the worker herself. Attention must be given not only to objectively observable work performance factors but also to subjective features of the worker's experience and actions. Such features can include (e.g.) idiosyncratic practices, tacit knowledge, rules of

thumb, private language, localized jargons, personal categorizations of subject matter, and the like.

Why are such subjective factors important? Information-intensive work is typically supported with automation, but it is rarely automatic. Whether the task be case processing or command and control, the presence of a human 'in the loop' means specific outcomes will be predicated on the particular actor's perceptual capabilities, cognitive capabilities, and praxial ("of or relating to praxis") capabilities. Actions evidence decisions, which in turn are based upon apprehension and interpretation of the subject matter at hand. The most common question posed in challenging a deliberate (i.e., decided) action is: "What were you thinking?" Accordingly, the focus for analyzing deficiencies in, and improving capacity for, deliberate action must be the subjective context within which the decision maker operates. This means the analyst's task is to learn enough about actual praxis so as to be able to comprehend, emulate, or even simulate worker experiences and orientations as they occur in the flow of the target work. Phrased another way, the goal of KA is to educate the analyst to be a surrogate subject matter expert. This is the wisdom in PD's emphasis on *mutual learning* in the design process.

Armed with this understanding, the analyst is finally ready to undertake design. However, 'design' takes on a different character when performed from a phenomenological orientation. Under this orientation, design becomes a matter of specifying what needs to be present to the worker (from her 1PP) during her performance of the target work activity. This will typically include specification of what must be perceived, what interpretations must be made to foster understanding, what representation or expression of this understanding best informs the actor, what must be decided based on this understanding, and what one must do to effectuate decisions. Unless the IS team is privileged to have worker participation throughout the project (the PD ideal), it will fall to the analyst / designer to derive these key elements by simulating the worker's praxis.

The flow of work experience becomes the foundation for prescribing what might be done. The target task's process path serves as the template for a figurative track along which an actor must proceed. This track is punctuated by points at which something shifts, begins, or ends. Each such discernible juncture recommends what must be perceived, interpreted, decided, and enacted before proceeding. The es-

sence of design is accreting that which recommends itself to a set of prescriptions for the new work environment's features and then creating a coherent model embodying these features. Regardless of the terms in which project motives were originally described, in this approach improvements are framed as re-definitions of the 'milieu' within which the worker operates and with which she engages. This emphasis on re-engineering the milieu underlies my inclination to characterize the intended outcomes of a phenomenologically or hermeneutically sensitive process as *interventions*.

At the point the designer undertakes to specify prescriptions for intervention, there is no presumption these prescriptions will include a novel IS artifact. Only after these prospective interventions are made apparent can they (and should they) be considered with respect to the need for, and the form and functionality of, a new IS artifact. Deferring commitment to a technological 'fix' minimizes the risk of unnecessary or misdirected software development. Avoiding premature technical development minimizes the biases induced by fixating on prototypes as unavoidable exemplars of the eventual product. Most importantly, patience allows generation of requirements specifications to proceed with respect to what the target worker needs to be made facile in the future rather than what functionality technological precedents have provided to date.

In other words, software prototyping is undertaken with respect to realizing the envisioned new milieu, and its features are circumscribed by the features that new milieu has been taken to incorporate. The primary criteria for evaluating the prototype are therefore framed with respect to 'utility' (how useful it is in facilitating worker actions) rather than how 'usable' it is in and of itself. This is not to say that usability is disregarded. It only means the issue of whether the worker can better engage *this* task takes priority over the issue of how well the worker can operate *this* IS artifact. Figuratively speaking, this orientation emphasizes how well a person in the role of a worker can drive a nail over how well a person in the role of a user can swing the hammer.

Because mutual learning is critical to pursuing this style of design and development work, allowance should be made for recurrently interacting with the target workers. This is done to both (a) learn more about the issues which will inevitably surface as the analyst simulates what it's like to do the target work and (b) obtain feedback and validation from the workers themselves. This requirement for a deep working knowl-

edge of the worker's first-person perspective effectively mandates organizing the IS agenda as a series of 'loops' rather than a stepwise linear progression. In my experience, additional time invested in this learning and consultation with the real experts on the target work will consistently be rewarded with efficiency in both (a) identifying the most constructive interventions to be pursued and (b) generating interventions satisfying the most needs on the first pass.

Summary Comparison of the Two Orientations

Table 1 summarizes some key points distinguishing the conventional IS design and development mindset from an orientation more accommodating of phenomenological and hermeneutic considerations.

Table 1: Summary Comparison of the Two Orientations

<u>POINT OR ISSUE:</u>	<u>CONVENTIONAL ORIENTATION</u>	<u>PHENOMENOLOGICAL ORIENTATION</u>
Focal Product of IS Development	IS artifact(s) to be employed by workers	A revised work milieu better accommodating and facilitating worker praxis
Intended Outcome of an IS Project	Innovation with respect to the form or function of the new artifact(s)	Intervention with respect to the worker's milieu of praxis
Ascribed Role of Humans	<ul style="list-style-type: none"> • Functional components of the composite work system • 'Users' of the IS artifact(s) 	Actors from whose perspective the work activity and subject matter are to be portrayed
Perspective Taken on the Joint Human-Machine Work System	Third-person (external; objective) vantage on both the work system and the human component(s) within	First-person (subjective) vantage of the worker with regard to her work activities and milieu
Referential 'Anchor' for Analysis and Design	The artifact is in the foreground as the fixed point of reference	The worker's experience is in the foreground as the fixed point of reference
'Variables' Presumed Mutable as Necessary	<ul style="list-style-type: none"> • User experience • User interpretation of data provided by the artifact 	<ul style="list-style-type: none"> • Artifact form • Artifact function

<u>POINT OR ISSUE:</u>	<u>CONVENTIONAL ORIENTATION</u>	<u>PHENOMENOLOGICAL ORIENTATION</u>
Focal Evidence to be Collected	<ul style="list-style-type: none"> • Quantitative performance data • Formal descriptions of work organization, process, and tools • Emphasis on ‘universals’ applicable to all workers 	<ul style="list-style-type: none"> • Qualitative accounts of praxis • Informal accounts of actual socio-technical interactivity • Attention to ‘personal’ or ‘subjective’ work issues
Presumptive Analyst / Worker Relationship	<ul style="list-style-type: none"> • Analyst diagnostically examines the subject worker(s) • Analyst accumulates sufficient data to formally model operations and constituent functions • Analyst prescribes an innovation based on the formal model 	<ul style="list-style-type: none"> • Analyst learns from the subject worker(s) • Analyst achieves sufficient understanding to simulate worker states in the flow of the work • Analyst describes an intervention based on discerned worker needs
Criterion for Evaluating Outcomes / Product(s)	‘Usability’ – How well the human user operates the IS artifact in and of itself – often without regard to the work itself	‘Utility’ – The degree to which the worker can engage work subject matter and execute tasks with minimal attention to the artifact itself

Lessons Learned in the Field

The preceding section discussed the general ways in which a phenomenologically or hermeneutically sensitive orientation results in practices distinct from prevailing conventions. This section will offer more specific illustrations of some lessons I've learned in pursuing phenomenologically oriented IS design. In the wake of the 1990 train conversation cited at the beginning, I found myself working to identify how – and how far – one could proceed in applying phenomenology and hermeneutics in practical IS design. In the years since I have reached some conclusions about both (a) the prospects for applying these ideas in theory and (b) some techniques for applying them in practice. The following sections will present what I believe to be the most important

things I've learned in attempting to move phenomenological and hermeneutic themes from academe to the workaday world.

The Context for Applying these Ideas

The first step in this exposition is to explain the IS design and development context within which I've been applying ideas drawn from phenomenology and hermeneutics. Since 1999 I have been involved in a series of projects aimed at developing decision aids for a command and control environment. These projects' research emphasis has been on creating visualizations and other interface components affording decision makers better focus on relevant subject matter and a resultant capacity for more efficient and effective decisions. Figure 1 illustrates the manner in which I've come to view the subject matter of such projects.

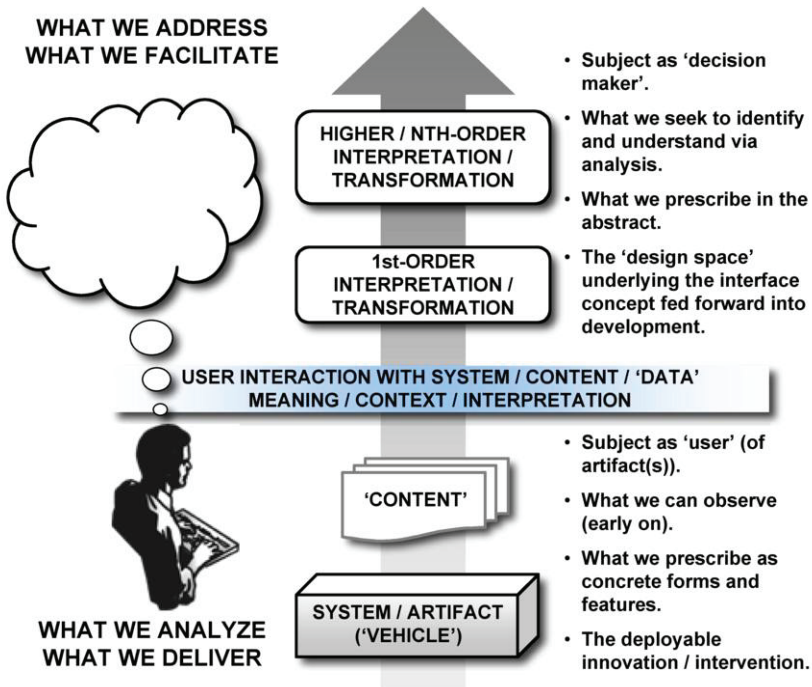


Figure 1: Overview of the IS Design Context

As illustrated in Figure 1, there is an important distinction to be drawn between the externally observable elements of the work milieu and those that are 'personal' to the target user / worker. This distinction is

analogous to Kant's division between *noumena* and *phenomena*. The directly observable components of the work milieu (artifacts, data 'content', user activity patterns, etc.) are the objects for which data is gathered in knowledge acquisition and explored during analysis. However, effort is invested in attempting to identify and describe those 'personal' means by which the worker engages her subject matter – e.g., interpretations or translations of observable data as those items are engaged in her personal cognitive context. The process of 'interpreting' observable data for meaningful effect on the target work process is the point at which hermeneutics is most applicable. The nature and utility of the resultant interpretations (as discrete items of reference in the worker's cognitive domain) - along with any permutations, transformations and other operations performed on or with reference to those interpretations - provides the area in which phenomenology is most applicable.

The point in doing this is to identify the elements most pertinent to the worker's first-person perspective on the subject work. Such elements do not necessarily reflect the categories, definitions, etc., laid out in those 'external' data sources (e.g., training materials, handbooks, etc.) upon which conventional analysts typically rely. Most important are those descriptive or interpretive elements that have arisen in praxis – i.e., referential items whose criticality has been identified by virtue of their recurrent importance in work performance. Examples of such personally generated and / or personally enacted elements include:

- Clusters or sets of subject matter elements routinely treated as wholes
- Routine sequencing of subtasks and evaluations of completion criteria
- Factors identified as indicators of problems or opportunities for task completion
- Cues indicating tangential requirements (e.g., for additional data)
- Key features used to judge states of the subject matter or situation at issue
- Relationships routinely checked among subject matter elements
- Qualifications or glosses applied to conventional work domain elements

As connoted in Figure 1, such personal ‘constructs’ may be ‘first-order’ transformations on observable data sources, while others may be ‘higher-order’ transformations building upon such first-order interpretations. Prioritization of such constructs for representation in the design concept is to some extent proportional to their role in the generation of such higher-order constructs. In other words, if lower-order personal construct X serves as the basis for higher-order constructs Y and Z, it is usually most critical to ensure X is reflected in the eventual design concept.

The goal is to establish as comprehensive a mapping of the worker’s ‘phenomenal’ or ‘cognitive’ operational domain as possible. Once this has been satisfactorily accomplished, the next step is to craft design requirements with primary regard to such ‘phenomenal’ factors. This closes a procedural loop starting with the observable work milieu, leading through examination of the ‘phenomenal’ projection of the work milieu, and ending with specification of novel observable / external work milieu components (e.g., IS applications). The intended result is an IS design that reflects the worker’s 1PP as much as possible – i.e., an IS design that emulates the worker’s usual ‘vantage’ on the work and the work subject matter.

I have applied this general approach to create a series of IS designs which have been accepted and moved forward to development and deployment. These projects’ outcomes have been sufficiently successful to motivate description and promotion of their origins in terms of both a new type of IS product (*work-centered support systems*) and a novel form of IS design practice (*work-centered design*) (e.g., Eggleston, 2003; Eggleston & Whitaker, 2002; Eggleston, Young & Whitaker, 2000; Scott et al., 2005). Grateful as I am for these developments, I cannot claim they reflect more than isolated tidbits about my actual design praxis and tactics in the course of the referenced projects. More importantly, none of them reflect the phenomenological or hermeneutic influences that to my mind underpin both my IS design praxis and its successes. As a result, I use the term *praxio-focal* to connote the praxis-centered nature of my methods and to prevent confusion with those aspects of the formally documented ‘work-centered’ approach appended to (rather than reflective of) the facts of my IS design experience.

The remainder of this paper will provide an illustrative overview of my praxio-focal IS design approach with specific emphasis on the key roles

phenomenology and hermeneutics have played in its formulation and application. In the following sections I shall more deeply discuss selected aspects of this approach and their relation to ideas drawn from phenomenology and / or hermeneutics.

The Scope of Applicability for these Philosophical Ideas

In some cases, phenomenology and hermeneutics have been touted as panaceas for reforming or improving IS development methodologies. Though these fields can certainly enhance IS design quality, some of the recommendations made for them are naively overblown. My experience indicates it is vital to understand these ideas' scope of practical applicability and to circumscribe their usage accordingly. This issue of scope has to be addressed in two distinguishable contexts: (a) the scope of IS project subject matter to which these approaches offer useful insights, and (b) the scope of IS project activities to which they can be expected to usefully contribute.

First is the issue of circumscription with respect to subject matter. My project experiences consistently support a point not often made in the literature promoting them - *phenomenology and / or hermeneutics do not and cannot represent a comprehensive alternative to conventional IS project methodology*. Owing to the fact these approaches concentrate on the individual human's cognitive processes, they are informative only when addressing the human worker or user – either in isolation or in relation to current and prospective IS artifacts. When considering other aspects or dimensions of the target work milieu, individual capacities and processes may decrease or recede in relative importance or relevance. The two most common such situations occur when attention is given to (a) social / organizational (i.e., supra-individual) aspects of the work activity and (b) purely technical aspects of the work environment.

Second is the issue of circumscription with respect to IS project itinerary. The foregoing discussion touched on only certain portions of the conventional IS project progression – knowledge acquisition, analysis, design, and culminating evaluation. The portion of the project stretching from design specification delivery to production of a prototype was not mentioned. The reason for this is that the software development phase is the phase least amenable to influence or improvement through application of phenomenological ideas. Any recourse to issues of (e.g.) apprehension or interpretation during this phase are likely to occur only

where the upstream project phases have omitted, overlooked, or misconstrued some aspect of the target worker's task experience. In other words, there is little likelihood of such issues being relevant unless the developers need something elaborated or clarified. In my experience, this most commonly occurs when developers question some aspect of the design concept and the underlying design rationale must be explained. The later evaluation phase (if undertaken at all) typically involves reference back to the design rationale developed in the early phases of the project. This means evaluation is usually concerned with phenomenological / hermeneutic factors identified and analyzed much earlier in the project. As a result, my experience indicates *the primary opportunities for leveraging phenomenological and / or hermeneutic ideas lie in the early phases of the IS project's process path.*

Circumscribing the Designers' Interpretive Problem

The prevailing (logical-empiricist) mindset fails to address, much less illuminate, subjective aspects of the worker's experience. Obtaining a good 'fit' between IS designs and workers' praxis requires that both data and the portrayal of such data be framed in such a way as to facilitate worker apprehension, comprehension, and decision processes. This would imply a need to analyze and reflect the each worker's phenomenal states during the course of the target praxis. Unfortunately, the analyst has no direct access to the worker's perceptions or thoughts. As such, elements ascribed to the worker's phenomenal domain are always objects of hermeneutic analysis on the part of the observer / analyst.

The observer / analyst's predicament can be readily illustrated with regard to the concept of *cognitive point of view* as described by Varela (1979, p. 85) – "... a particular set of presuppositions and attitudes, a perspective, or a frame in the sense of Bateson ... or Goffman ...; in particular, it is associated with some notion of value, or interest. It is also linked up with the cognitive capacities (sensory capabilities, knowledge background) of the distinctior." The observer's (analyst's; designer's) cognitive point of view circumscribes the particular 'layout' or 'topology' of her observing situation. This circumscription specifies the focus of observational engagement (i.e., where the observer's 'referential cross hairs' are targeted), and this in turn specifies the topology of the observer's immediately-accessible domain of referentiality. In other

words, the observer / analyst's capacity for addressing the subject worker is constrained by the observer's own phenomenology.

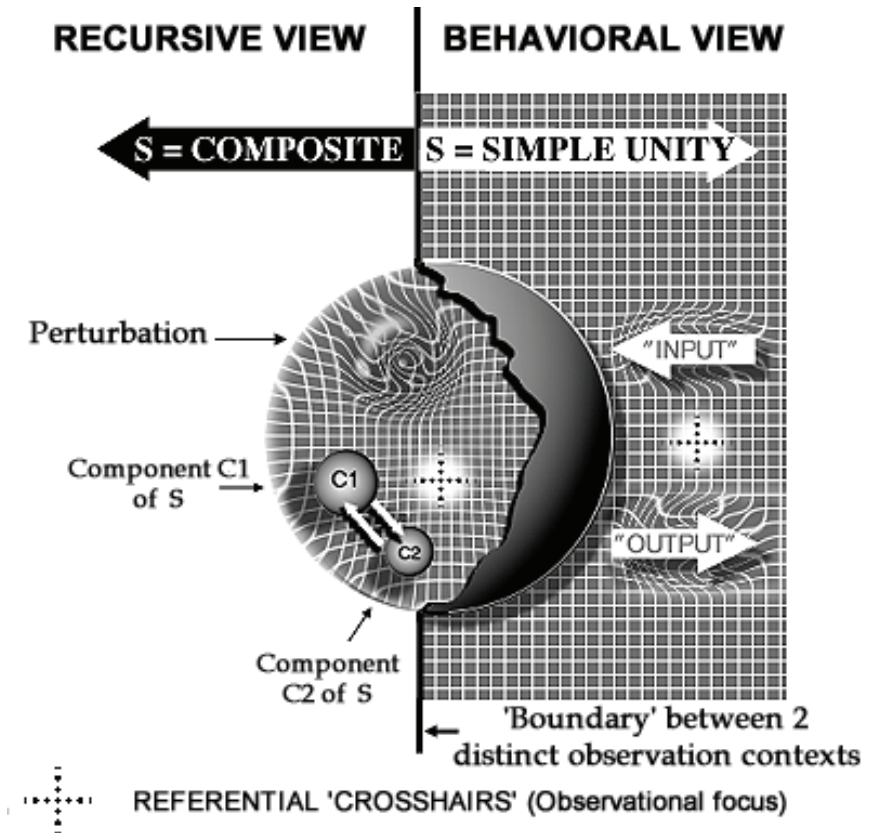


Figure 2: Adapted from Whitaker (1998)

Varela distinguished two basic cognitive points of view that can be taken on a given system. The *recursive view* is framed with regard to the operative components of the system, and the *behavioral view* is framed with regard to the interrelationship of the overall system and the milieu in which the observer observes it to operate. Adopting the behavioral view renders the subject system 'opaque' to inspection of its internal operations. From this perspective the system is an undifferentiated whole – a *simple unity* in Maturana's terminology (e.g., Maturana, 1978). Conversely, adopting the recursive view obscures one's ability to contextualize overall system actions in the subsuming milieu. This is because the context of reference is the set of elements comprising the

system seen as a *composite unity* (Maturana, 1978). This dichotomy is illustrated in Figure 2.

Conventional IS design methodologies address the user from a third-person perspective (3PP). From the 3PP vantage the user is a simple unity, and analysis is a matter of reducing her operations to an inventory of inputs and outputs (relative to the work activity; relative to the IS artifact). If one wishes to invoke the user's phenomenological or hermeneutical processes, it becomes necessary to attempt to emulate the user's 1PP, and this requires more of a recursive view. In the absence of any ability to observe the ephemeral elements comprising a worker's cognitive realm, this would seem an impossible task. As the next section describes, however, there is a means for focusing the scope of the observer's examination so as to obtain interpretive leverage on such otherwise intangible subject matter.

Action as Evidence of Phenomenal Processes

The major challenge in phenomenologically informed IS design is how the analyst / designer may reasonably ground the target worker's first-person perspective and experiences with respect to some form of observable or discernible evidence. To use the terminology introduced in the preceding section, the problem is what may be exploited from a behavioral view of the worker / user that affords approximation of a recursive view onto her phenomenal and interpretive processes.

The first step in attacking this problem was to accept the fact that there are distinct domains within which the subject person operates. This was the basis for delineating a set of *venues* (Whitaker, 1992) analogous to Maturana's *phenomenological domain* construct. First, the IS user operates in the role of a worker within a 'task venue' – the domain of observable work functions and operations. Second, this person operates in the role of a 'user' engaging the IS artifact via a 'depictive venue' in which elements of the work subject matter are presented for inspection and manipulation – the domain of (e.g.) on-screen data and representations. Finally, the person operates in the role of a 'phenomenal / hermeneutic operator' within her 'cognitive venue' – the domain of perceptual and cognitive processes. The first two of these venues can be reasonably addressed from a 3PP; the third requires attention to the subject person's own 1PP. The remaining issues lay in (a) obtaining tractable means for describing the cognitive venue and (b) interrelating the worker's tri-fold experiential context in a coherent way.

The depictive venue – as the object of an eventual design - is not available for exploitation in analyzing praxis or cognition. The cognitive venue is inaccessible by definition. This leaves the task venue as the basis for grounding analysis. Once again I turned to the work of Humberto Maturana, whose biologically-grounded phenomenological theory claims, "All doing is knowing and all knowing is doing" (e.g., Maturana & Varela, 1992, p. 27). Maturana's point is that what we colloquially term 'knowledge' is an allusive projection or fiction evidenced by effective action.

This entails a conversion from treating “this modeled knowledge as impetus for action” to “action as evidence for what may be construed as 'knowledge' “. Having adopted this converted view, I found that modeling the course of praxis is an effective representational foundation upon which I could plot my attributions (i.e., my interpretations) for the perceptions, interpretations, and decisions requisite to the series of actions demarcating the trajectory of that praxis. Only after contextualizing these ascribed ‘elements of information’ with respect to actual praxis could I begin to uncover situation-specific nuances providing better clues to both (a) what the work praxis really entails with respect to the worker’s phenomenal processes and (b) the dimensions of the phenomenal and hermeneutic domains within which the worker actually operates.

Correlating Praxis and Phenomenal Processes in a Coherent Model

It has proven difficult to identify a modeling schema configured to portray a combination of actions or activities in terms of elements that could be correlated with 'cognitive' or 'mental' (i.e., phenomenological) events. A usable example for what I sought was not to be found in the literature on work analysis, cognitive psychology, or other relevant fields. In most cases, the available models addressed the worker from a 3PP – treating her as a ‘black box’ and affording no features for addressing the worker’s ‘internal’ processes or experiences.

Eventually I found what I was seeking in, of all places, military science. This was the *OODA Loop* of Col. John R. Boyd (1987). The acronym stands for Observe - Orient - Decide - Act, and a 'loop' is a cycle comprised of these four phases. For all its apparent simplicity, Boyd's OODA Loop exhibits features, which recommend it as a schema for cognitive modeling. First, it explicitly addresses a decision / action cy-

cle in terms of continuous process from perception (Observe) through cognition (Orient / Decide) to instrumental response (Act). Second, the OODA Loop explicitly ties a system's perceptual / cognitive process to that same system's action toward its operational environment, and vice versa. Third, the OODA Loop can be used as a recursive construct in which any of the four phases can be decomposed into one or more subsidiary loop depictions. Plotting available data onto an OODA schema provides both a framework for organizing what I've learned and a basis for identifying what must be added to flesh out a useful model correlating what the worker sees, interprets, and decides with what she does.

Combined with my earlier theoretical explorations (e.g., Whitaker, 1992), this culminated in an integrated model for deliberative task cycles, as illustrated in Figure 3.

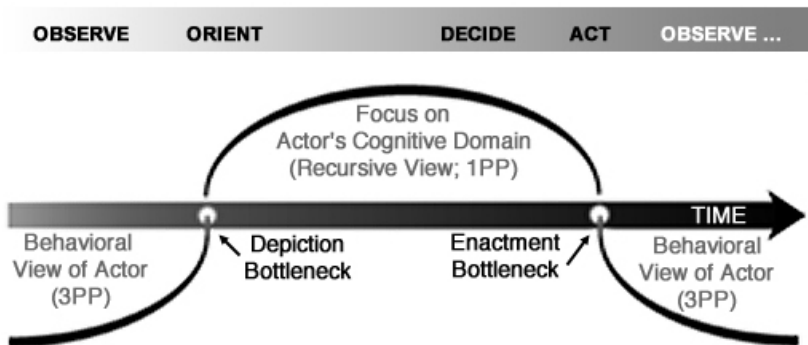


Figure 3: Process Schema Model for Deliberative Task Cycles

For any discernible cycle (whether an entire task or some constituent sub-task) the OODA progression can be punctuated into two primary sections or categories relative to referential focus and vantage. At the beginning and the end of the cycle the activities associated with the Observe and Act phases involve the ‘external’ work environment (which may include the data presentations afforded the IS user). In other words, these phases involve elements amenable to inspection from a 3PP. In the middle of the cycle the Orient and the Decide phases are primarily conducted with respect to the worker’s ‘internal’ cognitive domain. These intermediate phases involve elements directly

available only to the worker's 1PP – elements which the analyst / designer must infer and interpret.

The critical transitions between these '3PP-accessible' and '1PP-inferable' states occur during the initiation of the Orient phase and the Act phase. In terms of both (a) required effort and (b) potential for error, these transitions serve as 'bottlenecks' in the process flow. The 'depiction bottleneck' is the critical juncture from data apprehension (observation) to conceptualization and comprehension. The 'enactment bottleneck' is the critical juncture at which a decided course of action is put into effect. It is not surprising that these junctures entail risk, because it is at these junctures where the worker has to correlate her phenomenal / hermeneutic processes with states of the work environment (including tools and functions).

The IS designer seeks to constructively intervene into the subject worker's capacity for effective observation and initial orientation to the data observed by reconfiguring elements of the task venue. Intervention in support of the deeper aspects of the Orientation phase and the Decide phase is pursued through implementation of a depictive venue well-suited to portraying work subject matter in a manner consistent with the worker's discerned perspective, terminology, logic, etc. Intervention in support of the Act phase is pursued with respect to the task venue again. The designer is obligated to both (a) recognize the risks immanent at the depiction and enactment bottlenecks and (b) attempt to mitigate these risks by minimizing the cognitive burden attending transition at each juncture.

To employ the model above, the analyst / designer must work 'inward from the ends' in collecting and examining data on the work milieu and praxis. The data available for utilization in the Observe phase can be projected via a sort of forward-chaining inference process to delineate what it can support through the subsequent phases. Phrased another way, one works forward (through the process path representation) to circumscribe what can be supported later with what is available earlier. Conversely, the set of actions noted or inferred for the Act phase can be projected via a sort of backward-chaining inference process to delineate what their accomplishment entails as far as observed data, comprehension, situation awareness, etc. Phrased another way, one works backward to enumerate what may be required to decide and effectuate a given action. Both strands of inference and analysis are pursued in a process of recursive 'circulations' back and forth through the model.

As this process unfolds, the analyst / designer progressively weaves together a description for the informational and praxial elements comprising a workable environment for performing the target work activity. This description is the basis for designing an IS artifact whose form and function is geared to the worker's 1PP as she conducts the target work activity.

Obviously, such procedures mandate considerable effort and discipline on the part of the analyst(s) and designer(s). Owing to the focus on subjective factors, it is difficult to establish or exploit objective standards or 'universals' in framing analytical results or demonstrating value added. If anything, the phenomenologically-oriented analyst / designer must accomplish more than her methodologically conventional analogue, because she must both (a) generate analytical and design results plus (b) establish a comprehensible rationale for why these results should guide subsequent IS development. As such, *the burden of effort and responsibility for adequately accomplishing analysis and design activities is at least as great, if not greater, when adopting a phenomenologically or hermeneutically informed protocol.*

Illustrative Examples of Tactics for Fleshing Out the Model

Fleshing out the basic schema above is a laborious but creative task. Interweaving the descriptive data on the early and late phases of the process cycle(s) may be pursued in a number of ways. In this section I shall offer illustrative examples for three of the more useful tactics I've identified over the years. All these tactics prioritize clues obtained from the workers themselves or discerned in observations of worker praxis, because there is no substitute for engagement with the people who actually perform the target work.

The best clues for constructive interventions are derived not from what 'works', but from what doesn't. These shortcomings and pitfalls are typically evident only to the people who must confront them on a daily basis. Heideggerian 'breakdowns' are perhaps the most valuable clues, as evidenced by interruptions, problems, and digressions in conducting the target work. One quick observation of a worker 'stumbling' or performing a laborious workaround can be more valuable than hours of observation data on smoothly uneventful operations.

Because my modeling and analysis approach emphasizes the work process path, discerning the actual course of work activity is critical to success. A particularly rich source of clues lies in any personally- or locally-created work aids (checklists, 'cheat sheets', and the like) the workers have generated to support their own praxis. Where such aids have been developed, one can usually assume the work activity entails complexities that must be addressed or accommodated in the eventual IS design. Such homegrown aids can provide direct evidence of gaps or deficiencies in current work support, and in some cases I've found them to be directly convertible into well-accepted IT features and artifacts. In any case, a robust description of actual praxis gives the designer an inventory of steps, terms, and / or actions that should be reflected in the eventual design.

The analytical objective in this IS design approach is to identify what a worker knows (or needs to know) and when she knows (or needs to know) it in the course of her praxis. To correlate these key phenomenal or hermeneutic events with the process path I sometimes conduct what I call a 'horizon exercise'. Given a progression or stepwise activity being studied, I will generate a set of key information elements relevant to performing the activity - e.g., data to be obtained, analytical results to be generated, decision factors to be considered, and so forth. The subject matter expert is then presented with an outline of the activity's phases or stages (e.g., a series of open blocks on a whiteboard). I then go through the list of information elements and ask the SME to indicate at what point in the sequence she will or should know or need to know that particular item. Figuratively speaking, I am probing for the 'horizons' of information and knowledge traversed along the process path. This permits me to obtain evidence of correlations among information elements with respect to one or another situated state along the process path. This sets the stage for identifying (a) information requirements for each of the process path's stages, (b) dependencies of information elements on prior information and actions, and (c) clusters of interrelated information elements suggestive of interface contents.

As illustrated in this and the preceding section, I've compiled a set of frameworks and tactics capable of usefully supporting the most problematical aspects of phenomenologically-informed IS design. I make no claim that my own methodological repertoire is the only – or even the optimal – set of 'tools' for phenomenologically-informed IS design. My only intent is to make a more fundamental point – i.e., *it is possible to*

assemble a theoretical and methodological 'toolkit' consistent with emphases on phenomenology and / or hermeneutics. At this early stage of development, phenomenologically- / hermeneutically-oriented IS designers have not closed – and in many cases have yet to substantively address - the issue of whether their theoretical preferences can be transformed into practical methods. Only after such methods are more widely created, refined and tested will we have any basis for considering the feasibility and optimality of any 'toolkit' prescription.

User 'Phenomenology' as the Object of Design

Having now touched on selected themes I've applied from phenomenologically-oriented sources, I shall now revisit the earlier overview of my IS design work context and restate some points with more specific regard to these sources. From the worker's first-person perspective (1PP), work can be addressed as an unfolding series of problem solving incidents. Effective IS interfaces should aid a user in recognizing, analyzing and reacting to problems encountered in her praxis. The quality criterion for an interface design therefore becomes the degree to which it facilitates such problem solving. This requires the interface to provide everything a worker needs to perform the work at hand with no other extraneous or distracting features. In effect, the design process becomes oriented to inducing an improved 'phenomenological domain' (to use Maturana's term very loosely) for the work activity. This in turn requires the designer to tailor the interface to reflect the way(s) in which the worker may optimally (most 'transparently'; most directly) address problem situations.

My approach to generating such interface designs begins with conceptualizing what is needed to address a given problem situation or scenario. The set of such situations to be addressed will have been derived as results of the process path analysis and cognitive processes analysis work cited above. These analyses should also have produced an understanding of the praxis associated with, and the information elements required for, comprehending and managing each such situation. Using these analytical results as a foundation, the designer must then generate one or more interface concepts tailored to accommodate or support each phase of the associated OODA cycle.

For complex decision support tasks, the most important step in this process is identifying the problem's essential features or factors and then identifying the referential background or context most amenable

to coherently portraying them. In other words, this is a matter of specifying the optimal point of view for addressing the problem - a *vantage* from which all salient aspects of the problem are discernible. The vantage is typically designed as a central display, around which are arrayed tools through which the user can (a) manipulate the vantage display's contents and (b) perform functions necessary to addressing and resolving the problem. These peripheral tools comprise a *frame* for the vantage display. Because this design strategy progresses from problem to vantage to frame, it has been labeled the *Problem-Vantage-Frame* approach (Eggleston & Whitaker, 2002).

In effect, this strategy approaches design conceptualization with respect to the target worker's first-person perspective. I use the term 'worker' rather than 'user', because the perspective sought is not the one a 'user' has on the IT artifact in front of her, but rather the one a 'worker' must have on the subject matter of the work she is attempting to perform *through* the IT artifact. Ideally, such an interface design enables the user to effectuate the entirety of the associated 'problem solving' OODA cycle through one display. To accomplish this, the visualization components must be configured to support all the worker perceptions and interpretations necessary to understand the problem, to evaluate alternatives, and in some cases to project the effects associated with a candidate course of action.

Evaluating Success

As noted earlier, principles of phenomenology and / or hermeneutics are most applicable to the earlier stages of an IS design and development project. As the project proceeds, there arise two very interesting issues framed with respect these ideas' demonstrable value:

- *Design Concept Acceptance*: What – if any – effect does adoption of these principles have on acceptance of the resultant design concepts during the course of the project?
- *Formal Product Evaluation*: What – if any – effect does adoption of these principles have on the evaluation of the project's product(s)?

Earlier I noted that a smooth flow of events without breakdowns is relatively uninformative. As such, the good news from my experience (i.e., a track record of concepts consistently accepted and products consistently evaluated as 'good') is - perhaps ironically - bad news for any-

one demanding definitive proof of such an approach's value. This is especially true with regard to the latter question (regarding formal product evaluation). By the time a phenomenologically or hermeneutically informed design concept has been translated into a tangible IS artifact, many of the product's particular features are directly attributable to choices made during the development phase, and this can impede assessments of the degree and extent to which such upstream orientations uniquely affected downstream outcomes.

On the other hand, the benefits of a phenomenologically-informed design are to some extent subject to the same contextualization and testability as the benefits conventionally claimed and documented in IS development practices. If the design meets its objectives for providing a worker with a maximally 'transparent' window onto the subject matter of her praxis, that worker should be facilitated in performing her task more efficiently and more effectively. This means performance-centered metrics for measuring such improvements (e.g., reduced error rates; faster processing time per case) are therefore as applicable to this new approach as to traditional methodologies. In my experience, when formal quantitative evaluations of such design concepts have been performed the data has recommended their efficacy on the basis of such metrics.

For what it's worth, I suspect the points upon which one could claim more uniquely attributable benefits lie around the periphery of the specific task activity being addressed. By this I mean such points probably relate more to the worker's personal (phenomenal; hermeneutic) engagement with her overall praxial flow than to any specific task within that flow. Such benefits are best illuminated if the evaluation context is framed with respect to a 'recursive' (internal) perspective on the worker's performance rather than a 'behavioral' perspective from which her performance is framed with regard to inputs and outputs. In the case of decision-intensive tasks, such points include (e.g.):

- The scope of possible alternatives made available for the decision maker's consideration
- The depth to which the decision maker can explore and evaluate the features of a given candidate course of action

- The extent to which the decision maker can project and assess the outcomes and follow-on ramifications of a candidate course of action
- The facility with which the decision maker can interrupt her flow of praxis and resume again without losing situation awareness
- The facility with which the worker can maintain situation awareness over the current work stream (as contrasted with activity on one or another ‘case’ within that stream)

All these issues are conceivably addressable in quantitative terms. However, the scope and the fidelity of a simulation environment capable of supporting illustration and measurement of such factors typically exceed conventional projects’ aspirations or capacities.

In any case, the ascribed value of such an approach with respect to both concept acceptance and formal evaluation will be directly proportional to the degree to which actual workers / users are involved as evaluators. Because such concepts are tailored to fit and facilitate workers’ personal work experience, such merits as they may offer will be most apparent to the workers themselves. Just as project engagement is critical at the project’s outset (e.g., in knowledge acquisition), it becomes similarly critical at the close. In between those two junctures it is the responsibility of the analyst / designer to serve as a ‘user advocate’ to ensure the earlier insights are preserved in the form of the eventual products.

The most extreme case of worker engagement in my own experience also serves as my nominee for the most clear-cut illustration of what ‘success’ really means. The project was a research study intended to analyze an entire population (less than 10 in number) of certain planners and to devise concepts for better IS support tools. Weeks of preparation led to days of on-site observation and discussion with the workers in their daily workplace. I was able to collect worker-generated work aids (checklists, etc.) and to observe deficiencies causing breakdowns in their praxial flow. In the end I made a single summary presentation to the entire worker population accompanied by an even larger ‘secondary’ audience of operational managers, IS managers, and IT contractors. After presenting my analytical conclusions and design concepts, I asked for feedback and comments. The increasingly awk-

ward silence was broken when one after another of the workers spoke up to state I'd (a) understood their work, (b) identified issues even they hadn't recognized, and (c) offered design concepts whose merits they grasped at first glance. I left the meeting feeling extremely gratified and 'validated', and the research project was deemed a success.

Some 3 years later I was at the same worksite working on another project. I ran into one of the original subject matter experts from three years before, who had become a manager for the planners I'd studied. He told me he wanted to show me something and led me to his office. He logged onto his computer workstation and invoked the software application used by the planners. What I saw was my PowerPoint design illustration instantiated in functional software. My host advised me that he and other the planners had been so firmly convinced of my analysis and design's merits that they'd proactively lobbied their organization until the concept was built and deployed. Given the organizational setting, this was an extraordinary occurrence. Just as I'd argued years before, the interface allowed them to perform their complicated planning activities with maximum situation awareness and minimal errors – even under the duress of the largest workload surge the organization had ever experienced.

I have created praxio-focal concepts whose viability has been quantitatively demonstrated with statistical significance. I have created visualization concepts whose viability has been practically demonstrated by their perseverance across multiple generations of software applications. Nonetheless, my tears at learning that one concept's fate remain the validation I consider most incontrovertible. To my mind, the fact that my results motivated the workers themselves to champion the concept makes that one project the very definition of a successful intervention wrought by focusing on the subjective aspects of work praxis.

This admittedly anecdotal example is not the only one. Since 1999 there have been a total of five substantial decision support artifacts designed via this approach – all of which have been accepted by the target user community and moved forward to development and deployment. Though one may argue such acceptance doesn't constitute 'objectively demonstrable *proof*' that phenomenology and hermeneutics can constructively inform IS design, neither can one refute the fact these outcomes constitute *empirical* evidence for the claim that applying these ideas can aid the IS designer in 'doing IS design right'. As a practitioner, this latter fact is more than sufficient for my purposes.

Conclusion

The issues surrounding application of phenomenology and hermeneutics in IS design extend well beyond the selected sample presented in this paper. Still, this presentation should be sufficient to illustrate that phenomenological topics need not remain confined to philosophical debates. Both phenomenology and hermeneutics offer insights that can inform better IS design and development practices. The difficulty lies in developing practices based on these theories. I would claim I have achieved "meaningful use" of these theories in the sense that I've evolved tactics for accommodating their tenets in real-world projects and seen those projects' IS products be consistently accepted and deployed.

As stated at the outset, this has been a report of field experience in the course of my own quest to improve IS design through application of principles derived from phenomenology and hermeneutics. This paper represents "reflection on use" of these theories, thus hopefully completing an iteration of a hermeneutic cycle. To summarize, the results of this reflection include the following points:

- *Phenomenology and / or hermeneutics do not and cannot represent a comprehensive alternative to the entirety of conventional IS project methodology.* These ideas are of primary utility when addressing individual workers in the course of their work. They are not necessarily so illuminating with respect to (e.g.) purely social / organizational or purely technological aspects of the work milieu.
- *The primary opportunities for leveraging phenomenological and / or hermeneutic ideas lie in the opening phases of the IS project path.* These ideas pertain to the worker's subjective experience, and this experience is more relevant to initially understanding the problem rather than subsequently constructing a solution.
- *The burden of effort and responsibility for adequately accomplishing these earlier phases is at least as great, if not greater, when adopting a phenomenologically or hermeneutically informed protocol.* Simply waving one's hands and invoking academic terminology neither gets the job done nor convinces anyone in the workaday world that these ideas have practical merit. In

any case, analyzing a target worker's phenomenological / hermeneutic processes is at least as involved as analyzing her observable task procedures.

- *It is possible to assemble a theoretical and methodological 'toolkit' consistent with emphases on phenomenology and / or hermeneutics.* As illustrated above, I've compiled a set of frameworks and models capable of usefully supporting problematical aspects of phenomenologically-informed IS design.
- *It requires considerable effort and creativity to equip oneself with both theoretical and methodological toolkits for phenomenologically or hermeneutically informed design.* As illustrated by my own theoretical explorations, I found it necessary to range across multiple disciplines to locate useful concepts, theories, and models. These still had to be augmented with novel tactics developed over the years.
- *At least to some extent, the merit in applying these ideas can be demonstrated.* As indicated in the section on evaluation issues, adopting these philosophical inspirations need not mean abandoning rational evaluation of outcomes. However, it does entail confronting ambiguities concerning whether specific outcomes are uniquely attributable to the influence of these philosophical ideas.

In closing, let me return to the story of the 1990 train encounter. I never met that German hermeneutics scholar again, and I have no idea whether our conversation had any influence on him or his work. For my own part, I came away from that night believing the theoretical applicability of phenomenology and hermeneutics to IS design was 'good news', and that the next step was to explore the possibilities for transforming this theoretical prospect into practical benefits. As described throughout this article, I've invested over 15 years in such exploration. I attribute the bulk of my subsequent success as an IS analyst and designer to the insights phenomenology and hermeneutics afforded me.

References

- Boyd, J. (1987). *A discourse on winning and losing*. Montgomery AL: Air University Library, Maxwell AFB. Report no. MU 43947.
- Checkland, P. (1981). *Systems thinking, systems practice*. New York: Wiley.
- Docherty, P., Fuchs-Kittowski, K., Kolm, P., & Mathiassen, L. (Eds.). (1987). *Systems design for human development and productivity: Participation and beyond*. Amsterdam: North-Holland.
- Eggleston, R. (2003). Work-centered design: A cognitive engineering approach to system design. *Proceedings of the Human Factors and Ergonomics Society, 47th Annual Meeting*, Santa Monica CA: Human Factors and Ergonomics Society, 263–267.
- Eggleston, R., & Whitaker, R. (2002). Work centered support system design: Using organizing frames to reduce work complexity. *Proceedings of the Human Factors and Ergonomics Society 46th Annual Meeting*, Santa Monica CA: Human Factors and Ergonomics Society, 265 - 269.
- Eggleston, R., Young, M., & Whitaker, R. (2000). Work-centered support system technology: A new interface client technology for the battlespace infosphere. *Proceedings of NAECON 2000, Dayton OH*, 10-12 October, 499-506.
- Ehn, P. (1988). *Work-oriented design of computer artifacts*. Stockholm: Arbetslivcentrum.
- Greenbaum, J., & Kyng, M. (Eds.). (1991). *Design at work: Cooperative design of computer systems*. Hillsdale, NJ: Lawrence Erlbaum.
- Maturana, H. (1978). Biology of language: The epistemology of reality. In G. Miller & E. Lenneberg (Eds.), *Psychology and biology of language and thought* (pp. 27-64). New York: Academic Press.
- Maturana, H., & Varela, F. (1980). *Autopoiesis and cognition: The realization of the living*. Dordrecht: D. Reidel.
- Maturana, H., & Varela, F. (1992). *The tree of knowledge: The biological roots of human understanding*. Boston: Shambhala.
- Peirce, C. S. (1935). *Collected papers of Charles Sanders Peirce, Volumes 1 – 6*. Hartshorne, C., and Weiss, P. (eds.). Cambridge MA: Harvard University Press.
- Radnitzky, G. (1970). *Contemporary schools of metascience*, Göteborg: Akademiförlaget.

- Rasmussen, J. (1986). *Information processing, and human-machine interaction: An approach to cognitive engineering*. New York: Elsevier Science.
- Scott, R., Roth, E., Deutsch, S., Malchiodi, E., Kazmierczak, T., Eggleston, R., Kuper, S., & Whitaker, R. (2005). Work-centered support systems: A human-centered approach to intelligent system design. *IEEE Intelligent Systems*, 20(2), 73-81.
- Varela, F. (1979). *Principles of biological autonomy*. New York: Elsevier.
- Varela, F., Thompson, E., & Rosch, E. (1991). *The embodied mind: Cognitive science and human experience*. Cambridge, MA: MIT Press.
- von Foerster, H. (1981). *Observing systems*, Seaside, CA: Intersystems Publications.
- von Glasersfeld, E. (1984). *Radical constructivism*. Washington: Falmer Press.
- Whitaker, R. (1992). *Venues for contexture: A critical analysis and enactive reformulation of group decision support systems*. Umeå (Sweden): Umeå Universitet, ADB (Dept. of Administrative Data Processing / Informatics) dissertation / report UMADP-RRIPCS 15.92.
- Whitaker, R. (1998). *Encyclopaedia autopoietica*. Online reference guide to autopoietic theory concepts. Available at <http://www.enolagaia.com/EA.html>
- Winograd, T., & Flores, F. (1986). *Understanding computers and cognition: A new foundation for design*. Norwood NJ: Ablex.

Biography



Dr. **Randall Whitaker** is a senior human factors analyst with Northrop Grumman Information Technology, Vice-President for Electronic Publications - American Society for Cybernetics, and creator / manager of the Observer Web (the largest online resource on the work of Humberto Maturana and Francisco Varela).