Amethodical Systems Development: The Deferred Meaning of Systems Development Methods

Duane Truex*
GEORGIA STATE UNIVERSITY
Richard Baskerville
SCHOOL OF MANAGEMENT, BINGHAMTON UNIVERSITY
Julie Travis
CURTIN UNIVERSITY OF TECHNOLOGY
February 20, 1997
GSU CIS Working Paper
CIS-97-1
Amethodical Systems Development: The Deferred Meaning of Systems Development Methods
February 3, 1997

Duane Truex*
Computer Information Systems
College of Business Administration
Georgia State University
Atlanta, Georgia 30303-3083

Richard Baskerville
School of Management, Binghamton University
Binghamton, NY 13902-6015
Tel. +1 (607) 777-2337, Fax 777-4422, Email rbask@binghamton.edu

Julie Travis
School of Information Systems
Curtin University of Technology
Bentley 6102
Western Australia

All correspondence regarding this submission should be directed to Richard Baskerville

Abstract

This paper contributes a deeper understanding of the concept of methodical information systems development. The method concept is an assumption underlying much of the research into systems analysis, design and implementation. A postmodern deconstruction technique is used to discover an deferred concept: amethodical systems development. The methodical and amethodical views are developed in terms of their assumptions and their ideal characteristics. Our understanding of these two opposing views of systems development is important as a means to refocus our aims in research, practice and education in information systems development.

*This research was supported, in part, by a research grant from the College of Business Administration of Georgia State University.
Introduction

Do information systems development (ISD) methods really describe ongoing systems development practice? Do they ever explain why information systems (IS) are developed in certain ways? Do they actually function as frameworks, formulas or templates of successful ISD? Or are such methods merely unattainable ideals and hypothetical "straw men" that provide normative guidance to utopian development situations? Exploring these questions will help us to extract the essential lessons offered by the concept of IS methods. We address these questions by considering the alternative descriptions and explanations that arise when a hypothetical "method-less" view of ISD displaces our mainstream assumptions.

Methods for ISD are clearly important elements in the IS discipline. Yet there are gnawing problems about their practicability. Methods are often unsuitable for some individuals (Naur 1993\(^4\)) and settings (Baskerville, Travis and Truex 1992\(^3\)). Similar methods in similar settings yield distinctly different results (Turner, 1987\(^5\)). Developers may claim adherence to one method while ignoring this method in actual practice (Bansler and Bødker 1993\(^4\)). While ISD research has essentially reified methods, it offers little fundamental understanding of what it means to be methodical and how methods are actually applied in the field (Wynekoop and Russo 1993\(^5\)).

What does it mean for ISD to be methodical? Both Oxford and Webster’s dictionaries primarily define the term "method" as meaning "the procedure for obtaining an object." The secondary definitions fasten on such ideas as "orderly," "systematic," "regularity," and "regimen." Method is clearly a concept of process rather than representation. The term "methodical" is the adjectival form. In the mainstream discourse of our field, method is the term used for an orderly, predictable and universal approach to ISD.

"Methods are procedures which came to full flower in the validation of medical cures and in the development of pedagogical procedures and curriculum in the middle ages. Methods began as explanations but became procedures." (Ong, in Coyne 1995 p. 210)

What does it mean for ISD not to be methodical? We use the term amethodical to refer to this concept. The term is purposefully a negative construct that connotes an open set of attributes that are essentially not methodical. Thus amethodical may reject structure, but does not imply anarchy nor chaos. Amethodical ISD implies management and orchestration of systems development without a predefined sequence, control, rationality, or claims to universality. An amethodical development activity is so unique and unpredictable for each information systems requirement that even the criteria of contingent ISD methods are irrelevant.

The objective of this paper is to reconsider the concept of method; to think anew this key tenet in the ISD liturgy. The purpose is not to wreak ruin on the concept of method, but to clarify, extend and refine its meaning.

We extend our understanding of methodical development of IS by exposing the alternative observations, descriptions and explanations of the ISD process when considered from an amethodical viewpoint. This approach is necessary because the concept of method, as you will see in the following section, occupies an extremely privileged status in ISD thought even though its origin is unstated. This privileged position is so taken-for-granted in mainstream discussions and research about ISD that a technique was sought to allow an unfettered view of the method concept and alternative notions. Postmodern analysis uses such techniques for decentering the author of a text. The terms "narrative" and "text" are used in a metaphorical sense to encompass human artifacts such as organizations, a stream of human discourse, or social interaction, as well as written artifacts. Under this analysis, text is defined by the act of being read or observed, rather
than in being authored. The text's meaning is impermanent since changing readers implies possible changes in the meaning. A narrative is a body of texts, perhaps with some historical dimension, for example, the "narrative of science". A "metanarrative" regards a narrative about the narrative, for example, the philosophy of science is a metanarrative about the scientific narrative. "Local narratives" are bounded by a community of participants, for example, the local narrative of positivist science (Lyotard 1989⁹, Rosenau 1992⁹).

The paper will deconstruct method as a general ISD assumption (Eisenberg and Goodall 1993⁸, Kilduff 1993⁹, Beath and Orlikowski 1994¹⁰, Boje and Dennehy 1994¹¹, Coyne 1995¹²). Deconstruction is the process of disassembling socially constructed meanings that have been coded in texts (Eisenhardt 1989¹³). Deconstruction of assumptions in organizational texts reveals presumed structures in the reader's thinking, exclusionary conceptual frameworks (like method) that conceal multiple meanings. We use the deconstruction technique to confront the reader with conflicting meanings whereby any resolution assumes only a temporary dominance until overturned by the continuous process of reinterpretation. (The appendix briefly explains several related postmodern ideas for those readers less familiar with deconstruction.)

Essential to understanding our deconstruction is Derrida's concept of différance. Différance simultaneously connotes the differences in word meanings and the deferral of meaning through the endless play of word signs. Texts tend to be centered on an idea that is called "privileged" or "dominant" (e.g., method in ISD texts). The opposing idea may not be named in the text, yet exists as a shadowy, tacit concept that lingers in the reader's mind. This concept is referred to as the sub-text, hidden text or marginalized text). The exact meaning of this marginalized text is deferred to the reader (e.g., describing a method suggests that the reader understands amethod). This deferral sets up a cycle of redefinition that destabilizes the privileged text as a consequence of highly contextualized marginal texts (Cooper 1989¹⁴). Différance also implies that meanings are affected by the simultaneous understanding of the word and its marginalized anti-concept. Deconstructing a text involves seeking différance in order to consciously raise the concepts not openly discovered by readers.

Any text always involves sets of privileged and marginalized concepts. Concepts can be marginalized through secondary streams of signification. These streams involve the constant recursion and renegotiation of word meanings without ever reaching a transcendental ideal (discussed further in the appendix). Thus any linguistic artifact may ignite not only a primary thought, but an endless cycle of secondary references in a reader's mind. Différance also points to the impossibility of precisely defining any term without implying a delineation of all concepts to which the term must not refer. Since such precision requires the precise definition of infinite universe, which is impossible, all terms must be left to a certain degree, ambiguous.

A central problem in deconstructing ISD is the lack of any firm understanding of the marginalized concept of amethodical ISD. At the broadest level, this involves illuminating the marginalized text in the ISD metanarrative. Rather than deconstructing a single method as if it were a representative text of all methods, we will limit our scope to the discovery of method-amethod différance in our assumptions about the tradition of method that characterizes ISD research.

The second section will explain how the method concept fills its privileged role in the ISD text. Section three deconstructs the privileged text of ISD method and defers their meaning by giving voice to the marginalized amethod text. Section four continues the deferral, examining the cost of raising a formerly marginalized text (amethod) to the status of a privileged text. Section five concludes the paper by drawing the implications of the deconstruction for the important questions posed in the opening paragraph of the paper.
The privileged position of methodical ISD

The methodical view is privileged because the modern concept of method has been so strongly impressed on our thinking about ISD, that the two concepts, ISD and ISD method, are completely merged in ISD literature. We can quickly make this merger evident by exposing method as the privileged text dominating our publications about the history, practice, culture and philosophy of ISD:

**History.** Method is privileged because the history of ISD is typically interpreted as the history of ISD method. For example, modern analysts characterize the progress of ISD in terms of historic generations of ISD method beginning with Couger’s (1973\(^1\)) three-generation scheme (revised to five in Couger 1982\(^2\)) that links generations of ISD method to generations of computing hardware. Nolan (1979\(^3\)) examines six stages of organizational systems development progress in terms of the qualities and effectiveness of ISD method. Yourdon (1989\(^4\)) refers to three broad periods in the evolution of ideas and techniques in the field of systems analysis by focussing on use and nature of ISD method. Hirschheim and Klein (1992\(^5\)) delineate seven generations in terms of ISD method.

This powerful historical alignment leads to problems in distinguishing method from ISD altogether. Some analysts are compelled to redefine the term “method” beyond its original meaning, extending it to describe the discipline of ISD rather than the way systems are developed. These analysts conflate concepts like beliefs, values or material resources with method, often for the purpose of contrasting less methodical aspects of ISD such as hermeneutics or emancipation within the context of ISD.

The predisposition to believe in the power of methodologies comes from Descartes who proposed that truth is more a matter of proper method than genial insight or divine inspiration.” (Hirschheim, Klein and Lyytinen, 1995\(^6\), p. 21)

From the basis of such extended definitions of method, it might be argued that the privileged text as presented here is only one of many views of method. However, this privileged text is clearly the mainstream, technical-rational view (Lyytinen 1987\(^7\)). As a result, these other, broader views of method become marginalized along with the methodical.

**Practice.** Method is the privileged text among academic and practical communities involved that link to form the context for empirical research into ISD. As a result, much of the instrumentation that researchers assemble for the purpose of gathering data about practical ISD are only capable of detecting and measuring methodical concepts.

For example, Necco, Gordon and Tsai (1987\(^8\)), designed a survey instrument for determining how systems were developed. The respondent could report their ISD approach by selecting from a choice of alternatives that included traditional/classical approach, systems development life cycle approach, structured approach, automated approaches, etc. A setting in which a unique systems development approach is invented for each project could not be measured by this instrument.

Even if the instrument included such an option, the after-the-fact rationality of developers would marginalize the methodical. Developers are known to fake the rationality of ISD in their attempts to make sense of the process (Parnas and Clements 1986\(^9\), Bansler and Bødker 1993\(^10\)). Thus, empirical observations based solely in the reflections of practitioners would not detect or measure amethodical ISD. For example, Sabherwal and Robey (1993\(^11\), 1995\(^12\)) develop and analyze a rigorous inventory of ISD events that are solidly grounded on qualitative data from a broad set of interviews with IS developers. But this data only captures the actors’ rationalized reconstruction of the ISD events. An unplanned, on-the-fly collaboration that simply materialized in an uncontrolled setting might be characterized as a "committee" by an interview respondent in
an attempt to verbalize a sensible account of their project. The interviewer would record that "a committee was formed", which might later be assumed to mean that "someone formed a committee" (Robey 1995, p. 311) by the analysts. In such ways, our method paradigm prevents our research into ISD from discovering any practices beyond the methodical.

**Culture and Philosophy.** Method is privileged such that even philosophical and cultural analyses of the ISD milieu must be expressed in terms of ISD method. For example, intensive studies of ISD using Burrell and Morgan's (1979) framework inevitably characterize ISD paradigms according to the dominant ISD method (cf. Oliga 1988, Hirschheim and Klein 1989). ISD cultures are expressed in the regionalization of some ISD methods such as MERISE (Rochfeld and Tardieu 1983, Calmes, Charbonnel and Dumas 1991) in French cultural dominions and Soft Systems (Checkland 1981, Checkland and Scholes 1990) in British and Scandinavian cultural dominions.

Even our attempts to break free of the privileged confinements of ISD method are constrained to consider only "situated" ISD methods. For example, contingent systems development methods consider only the correct selection of methods with a comparative analysis of alternative ISD methods (cf. Avison and Fitzgerald 1983, Davis 1983, Jackson and Keys 1984, McFarlan 1981). Conceptualizations of idiothetic approaches are limited to "one-shot" ISD methods and meta-methods (Kumar and Welke 1992) for developing perfected one-time ISD methods. Almost entirely eluding the ISD literature is the thought of developing a system without any method at all (cf. Baskerville, Travis and Truex 1992). Even more elusive is the possibility that methodological ISD might be the normal way in which ISD actually occurs in reality.

**Methodical versus Amethodical ISD**

In this section we illustrate the distinction between the privileged methodical ISD text and the marginalized amethodical ISD text by selecting examples of major characteristics of ISD that are common in the privileged text. Then we draw the *différence* contrast for each example, and show how this explains reasoned and observed anomalies in the ISD process.

Different ISD methods require different techniques, sequences steps and activities. Nevertheless, there are common assumptions in these approaches which may be delineated as a privileged narrative. To the extent that the privileged narrative is dominant, any examination and greater understanding of the marginalized narrative is deferred. This section explores several key components of both the privileged and marginalized narratives of method. 0 summarizes the assumptions and idealized characteristics discovered in each of these texts.

*Privileged: ISD is a managed, controlled process*

One of the major assumptions advanced through the methodical narrative is that ISD is effectively managed and controlled by system developers. Methods provide universal mechanisms for achieving this management and control. Logical decomposition is one such mechanism; a "divide and conquer" techniques which decomposes uncontrollable wholes into a series of smaller controllable processes. Methods lead the developer through this reductionist process until each subcomponent becomes conceptually manageable and the construction resources become determinable and manageable. This decomposition further assumes that subsystems or components are interdependent slices of a system boundaries with independently stable and robust and yet (*i.e.*, highly cohesive and loosely coupled). The boundaries and relationships of modules, objects, *etc.*, are products of a rational design process. There is also an economic undercurrent in this assumption: Methods permit developers to "engineer" information flows in an effort to
improve the economics of labor and other resources in the production and maintenance of

Table 1. Assumptions and ideals of methodical and amethodical texts.

<table>
<thead>
<tr>
<th>Privileged Methodical Text</th>
<th>Marginalized Amethodical Text</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISD is a managed, controlled process</td>
<td>ISD is random, opportunistic process driven by accident</td>
</tr>
<tr>
<td>idealizing</td>
<td>idealizing</td>
</tr>
<tr>
<td>• logical decomposition</td>
<td>• holism</td>
</tr>
<tr>
<td>• reductionism</td>
<td>• creativity</td>
</tr>
<tr>
<td>ISD is a linear, sequential process</td>
<td>ISD processes are simultaneous, overlapping and there are gaps</td>
</tr>
<tr>
<td>idealizing</td>
<td>idealizing</td>
</tr>
<tr>
<td>• temporal causal chain</td>
<td>• opportunism and accident</td>
</tr>
<tr>
<td>ISD is a replicable, universal process</td>
<td>ISD occurs in completely unique and idiographic forms</td>
</tr>
<tr>
<td>idealizing</td>
<td>idealizing</td>
</tr>
<tr>
<td>• generalization</td>
<td>• choice</td>
</tr>
<tr>
<td>• consistency</td>
<td>• change</td>
</tr>
<tr>
<td>• formalisms</td>
<td>• adhocracy</td>
</tr>
<tr>
<td>ISD is a rational, determined, and goal-driven process</td>
<td>ISD is negotiated, compromised and capricious</td>
</tr>
<tr>
<td>idealizing</td>
<td>idealizing</td>
</tr>
<tr>
<td>• goal predetermination</td>
<td>• conflict</td>
</tr>
<tr>
<td>• process predetermination</td>
<td>• social constructivism</td>
</tr>
<tr>
<td>• human cooperation</td>
<td>• human independence</td>
</tr>
</tbody>
</table>

The history of ISD is dominated by methods that idealize and extend this "control by reduction" view of ISD activities (cf. Boland 1979⁴⁰, Kraft and Truex 1994⁴¹, Klein and Kraft 1994⁴², Hirschheim, Klein and Lyytinen 1995⁴³). For example, structured systems development (e.g., Yourdon and Constantine 1979⁴⁴), shifted the focus of decomposition and boundaries from physical computing machinery to logical software processes. Information engineering (e.g., Finkelstein 1989⁴⁵, Martin 1990⁴⁶) broadened the scope of systems design to an organization-wide scale, expecting a stable collection of data and processing components that would be shared by all organizational activities. This set of components would only need rearrangement and extension in order to adapt and grow. Object-oriented design (e.g. Shlaer and Mellor 1988⁴⁷, Coad and Yourdon 1991⁴⁸) broadens the scope still further, seeking stability and development control through reusable and maintainable components across, as well as within, organizational boundaries.

Even methods that do not directly regard functional decomposition assume reduction as a
control framework. For example, rapid prototyping (e.g., Naumann and Jenkins 1982, Carey and Mason, 1983) introduced a new means for controlling and stabilizing user requirements by iteratively refining components for user experimentation. Prototyping assumes that each iteration will produce a reduced set of problems for which permanent solutions can be fixed. As the design progresses, the problem set is steadily reduced to a level where specifications can be fixed and the final system implemented in a manageable fashion.

Marginalized: ISD is random, opportunistic process driven by accident

The privileged text elevates ISD as managed and controlled process, marginalizing the assumption that, instead, ISD might be a natural outcome of a complex, multivariate setting affected by many uncontrolled events. This marginalized text would assign method a minor, perhaps irrelevant, role in the ISD process. In order for methods to actually control ISD, developers must hold nearly perfect knowledge of a throng of interrelated factors. Many of these factors arise in the uncertain, confused arena of social behavior and autonomous human action, making predetermination impossible. Reductionist decomposition is problematic because the isolation of components is an imaginary process, and the resulting components do not reflect the rich interdependencies in organizational reality. Consequently the control is largely imaginary as well. The methodical actions join the mass of other irrational human activities that influence the ISD outcomes.

The marginalized text explains many anomalies of the ISD process, essentially the routine failures of the reductionist control. Many methods resort to a "fix-it-up" stage where the developers are given license to holistically repair the irrational and unacceptable designs that proceeded from the application of the method. Examples reach across the history of methods. DeMarco's (1979) version of structured analysis included a holistic reconsideration and revision stage. Data base designs require "de-normalization" in order to convert fully normalized designs into feasible specifications (Loomis, 1987). Object-oriented designs need rather amethodical adjustments to combine and separate perfectly rational objects to meet myriad unexpected factors, such as the politics of object libraries (cf. Coad and Yourdon, 1991). Methodologists, couched in the privileged texts, view such repairs as momentary, pragmatic escapes from method. The marginalized, amethodical text adopts an opposing view: This rather amethodical reconsideration of the outcome of the process is necessary because the assembled, imaginary result simply doesn't correspond with reality. The useful ISD work only happens in these repair stages, and in clandestine "extra-method" activities.

For example, Parnas and Clements (1986) noticed that IS developers produce documentation which only makes it appear that they followed a rational ISD method. The reality of the design process is a tortured discovery process, and the faked documentation disguises the way simple truths emerged. Much of the really useful information (like why alternatives were rejected) is not recorded:

"We will never find a process that allows us to design software in a perfectly rational way. The good news is that we can fake it. . . . The process is 'faked' by producing the documents that we would have produced if we had done this the ideal way." (Parnas and Clements, 1986, pp. 251 and 256)

The marginalized text explains why this amethodical vision of ISD better describes reality than the privileged methodical view. ISD is an art and not a science. Like any other process of creation, it is a process of discovery rather than design. It is open to serendipity and chance occurrences that cannot be predicted nor managed. This assumption recognizes that systems develop at their own pace and react to unexpected opportunities with constant and subtle adjustments that make the development project seem to have a life of its own.
For example, prototyping approaches expect user satisfaction to increase as an interactive prototype progresses. The set of user problems should decrease steadily. The privileged text assumes that the specifications steadily improve their alignment with user needs. These assumptions marginalize the possibility that users quickly weary of the process (Alavi 1984⁶), growing alienated or frustrated, and becoming less inclined to pursue prototype problems. This process also marginalizes the adaption of the users to the prototype, a circumstance in which the users must force-fit their world onto an imaginary setting required by the method.

Privileged: ISD is a linear, sequential process

Another fundamental assumption that inhabits the privileged text involves the straightforward linearity of methodical ISD approaches. Most ISD methods consist of ordered stages or grouped activities bounded by specific events or completed intermediate products. Indeed, the linear sequence of process (one step following another) provides much of the order and regularity that comprises the defining characteristic of the concept "method".

The practical problems with such temporal sequences were early recognized. These are compensated with iteration, feedback and backward-loops that enable repetition of earlier stages and events, and the revision of intermediate products. Even the waterfall model was quickly revised so that, in the case of ISD, water was sometimes allowed to fall "up" a little (cf. Yourdon 1989⁵⁰).

In fact, this iterative allowance relaxes only slightly the linear assumptions about ISD. The privileged assumptions about the causal nature of these sequences is largely undisturbed by iteration. Method assumes a progressive nature of the ISD process that rarely permits ISD to skip stages, events or intermediate products. ISD is a causal chain. The temporal importance of the sequence-progression of remains firmly anchored. While some steps may need repeating, none may be skipped or rearranged before its time. For example, step three (e.g., develop performance specifications) may be repeated several times, but never before step two (e.g., requirements determination) has been completed at least once. The intermediate products of a previous stage are usually assumed to be a precondition for initiating a following stage. Despite the presence of iteration, a tightly-dependent and temporal causal chain is assumed by most methodical texts.

The staging sequence partly justifies a methodical approach. ISD methods serve to organize activities. By ordering phases or steps as the sequence of activities, developers improve efficiency by eliminating excessive rework and in-process changes. Further the causal chain of ISD methods eliminate irrational activities, such as completing a design element before other elements on which it is dependent. For example, a methodical approach might proscribe the designing of the physical file layout before the analysis of the logical database elements, since the results of this analysis are likely to cause a revision of the file design.

Marginalized: ISD processes are simultaneous, overlapping and there are gaps

If the ISD process is random, opportunistic and driven by accident, it is unlikely that ISD could be marshalled along as a linear, sequential process. Amethodical ISD is an outcome of myriad development activities that emerge more-or-less independently. IS developers become engaged in the ISD activities that arise as the project evolves. The events that determine the real ISD sequence of activities are extremely complex and opportunistic. Examples of far-ranging factors that could affect the sequence include the preliminary expertise of the IS developers in the user problem domain, the availability of development machinery, the availability of users for interviews, and the technical expertise of the IS developers.

For example, Coad and Yourdon (1990⁸) observed analysts and designers at MCC in Austin Texas and noted that these professionals tended to work opportunistically rather than
methodically. The designers would work at a higher level of abstraction, then see a detailed area, dive into it, investigate it thoroughly, and then return to a higher level of abstraction. In fact, the ISD process is not a sequence at all. Many co-dependent activities continue simultaneously with varying degrees of communication. Stages get skipped when project schedules slip and hard deadlines approach. Some events are felt to be very critical and will be performed as soon as the opportunity arises rather than in the idealized sequence. A particular ISD project, or some of its activities, may stop altogether for possibly lengthy periods. Interruptions and changing developer expertise lead to unexpected revisions and improvements. It is altogether possible that the ISD process will produce an entirely different system with totally different purposes than those originally intended. The ISD process may also fade away without anyone taking particular notice that it entirely failed to achieve any closure. Not only must the rationality of the development process must be faked, but the original intentions must be revised to conform to whatever the outcomes must be.

The privileged text, with its methodical assumptions, obscure recognition of amethodical development. Any causal chain of dependent intermediate products in an ISD project is largely imaginary. Any one of the chain of products can be created first, and all others constructed for compatibility. For example, if the project appears to the developers as primarily a data problem, they may choose to grapple immediately with the data analysis and design regardless of any methodical sequence. Afterwards, any "preceding" intermediate products are created in a fashion that is compatible with the "subsequent", "dependent" data design. The ideas about product chains and intermediacy are not very relevant to ISD.

In fact, the assumption that exactly one activity must be first is methodical. The marginalized amethodical text assumes that several activities may begin simultaneously. Compatibility of activity outcomes is not really dependent on their sequence, but rather on cooperation, communication and negotiation. Many incompatibilities arise from this process (an frequent observation from reality that is easily explained by the marginalized text, yet poorly understood by the privileged text).

Privileged: ISD is a replicable, universal process

Methods have a surface appeal because they can seem completely mechanism, taking highly objective descriptions of organizational requirements and inescapably manufacturing essentially identical successful systems. This text assumes that adherence to method allows greater replicability of a system and that the method can be simply applied in a variety of organizational settings. An important ISD method is robust in the sense that it leads to dependable IS solutions in different organizational settings. A robust method can be generalized across a wide population of ISD problems. The object-oriented metaphor of the cookie-cutter (method) and the cookie (a successful ISD outcome) may apply here.

As was the case with the temporal sequence assumption, the privileged text is usually intemperate in its assumptions about universal application. Methods must be adaptable to a degree that allows their use in a variety of settings. The exact process may have to be preconfigured according to the needs of the exact ISD setting. However, the basic methodical framework will remain as an outcome of this reconfiguration. The adapted method is always a recognizable descendant of the original. This retained framework is avoidable because, unless the methodology ignores the adaptability issue altogether, it will usually attempt a predetermination of the adaption mechanisms. This privileged text assumes the preeminence of the methodical concept, thus a "meta-method" for adapting the method unavoidably emerges.

A typical mechanism for adapting methods arises from contingency theory. The exact ISD method is contingent on certain factors. For example, McFarlan (1981) shows how the preexisting IS application portfolio and the political situation of the IS organization might
determine the general approach to ISD. In another example, Davis (1982) uses various organizational factors to determine which ISD approach is best for a particular problem. Internally, many methods provide interchangeable components that can be selected contingently. For example, IS developers using structured methods may choose a data-oriented or a process-oriented approach contingent on the nature of the problem setting which involves a reorganization of the sequence (Yourdon, 1989). Interchangeable notation is often contingent on existing organizational preferences. For example, object oriented methods may permit object behavior to be defined with either pseudocode, flowcharts, or state diagrams.

At a fundamental level, methods achieve a degree of universality by being more-or-less formal. There is an assumption that the method has been carefully determined and its use in many situations is known to be successful. Deviation from the method is dangerous because failure will follow unless the alterations are very carefully devised. Adherence to a method will lead to consistent, successful outcomes while ad hoc approaches are hit-and-miss.

The formality is similar to the objective formality of scientific positivism. Scientific methods permit a universal approach to a wide family of problem situations, reduce these problems to an abstract set of symbols (often a diagram), and usually permit elements of the problem space to be symbolically manipulated with a finite set of operations in order to deductively arrive at the solution (cf. Gause and Weinberg 1989). At a functional level, the ISD methods introduce structure into the systems development process and regulate activities. These methodical structures provide ISD with a taxonomy of activities. This reduces the complex set of systems development activities using a formal classification scheme. Methods help achieve economies by reducing redundant activity, which become apparent when similar activities are grouped. Further, the formal element of ISD methods assure that activities are comprehensive. Formal rules insure that all necessary work is completed without oversight and eliminate the expense and embarrassment of returning to the project for additional "repair" development after the project's completion (cf. DeMarco 1982, Yourdon 1989).

Marginalized: ISD occurs in completely unique and idiographic forms

Amethodical ISD assumes that every information system emerges in its own peculiar way. The impact of vastly different variables in the ISD setting demand a completely unique IS process. These variables include organizational aspects like structures, cultures, resources, constraints, etc.; plus individual aspects like talents, habits, predispositions, etc. Organizations are too inconsistent to permit replicable treatments. Each ISD outcome arises in a revolutionary new way, not because a methodical framework has been adapted to the setting, but because an entirely new framework was invented for the setting.

One problem that inhibits our understanding of ISD is our predisposition toward the privileged text of the methodical. Our rational, faked explanations of the ISD process fail to note how the predetermined methodical framework collapsed during strategic events and the activities progressed in a manner incompatible with the espoused methodical framework.

Rather than providing a replicable and universal process, methodical frameworks (if attempted at all) are effectively discarded early in the process. Consistency in an approach to ISD is characteristic of IS developers who choose to mindlessly ignore the really difficult organizational issues that confront them. These developers absorb themselves in their own technology and rarely achieve more than mediocre solutions to the least significant aspects of the organization's needs. Amethodical ISD is the outcome of the exercise of unprogrammed, independent human choice during every ISD activity. Ad hoc approaches and methodical approaches are both hit-and-miss, but if the methodical approach is truly consistent, it consistently misses.

Fortunately, the predisposition to use "the same approach as last time" is typically abandoned early in a development process because the practical setting does not permit the
processes to develop. ISD methods are not adapted using contingency techniques. Rather, new components are invented altogether on-the-fly, and the unique framework, while always present, may never even be identified. IS developers may claim that certain contingent components were adopted and the framework followed. However, close examination may reveal that the actual activities do not resemble any of the contingent alternatives, and they were highly inconsistent with the claimed framework.

This marginalized assumption explains some observed problems with ISD methods. Bansler and Bodker (199365) report a triangulated research project in which they discovered that surveys reporting the heavy use of structured analysis in the workplace can be subsequently disproved by in-depth case studies. They first used questionnaire surveys to determine the ISD method used by an organization. This was followed by a case-study approach in which they observed the actual application of the ISD method. They found that despite initial claims that a particular structured ISD method was used, the more intensive investigation showed otherwise. Tools from the structured ISD methods were adopted, but these are combined in unique ways with other tools. Importantly, the structured procedures are not actually used at all.

The amethodical viewpoint also assumes that few organizations undergoing redevelopment of their IS will actually hold still during the process. Organizations are themselves emerging in a fluid process of internal change and environmental adaption. Any longer ISD process must itself emerge and change or it will slowly become irrelevant to the needs of its client. The contingencies cannot be predetermined because the nature of the organization at the conclusion of the process will be substantially different that the nature of the organization at the outset. Consequently, many of the ISD activities, events and products have to be reinvented along the way. Perhaps even the overarching ISD framework will transform completely.

This marginalized assumption explains why early ISD activities may be rejected during later phases. For example, Baskerville, Travis and Truex (199266) presented two case studies as evidence that long-term ISD methods are hopelessly ineffective because the organization changes quickly enough to invalidate initial findings. Consequently the later stages of ISD become inflicted by largely artificial structures that interfere with the development process. In this study, ISD methods are either subvertly or overtly abandoned and more pragmatic, locally invented approaches are adopted to complete the ISD.

The commotion in the underlying organization makes formal aspects of ISD methods problematic. Early intermediate products, like requirements specifications, are rendered obsolete by time lapse. Predefined budgets obviate the expensive rework. Besides, the time required for the rework may leave subsequent intermediate products equally obsolete. In the end, the ISD outcome convulses into a (perhaps successful) conclusion with every component, intermediate product and activity either discarded or simultaneously reworked. This is why an IS systems analysis is never completed, it just runs out of time (DeMarco 197967).

In fact, the nature of problems being presented to ISD have changed in the latest decades. Formal ISD methods have been used to cleanly solve problems that are amenable to formal methods, and over the years this has left a residue of "messy," less-amenable problems (Gause and Weinberg 198968). The early successes of formal ISD methods has led to boundary extensions of the problem domain to include a new target population of problems. Now the problem space for formal methods has become one that is dominated by problems which are highly unsuited to such formalisms. These problems are fraught with volatility, exceptions, unstructured data and unpredictable processing requirements.

This marginalized assumption also helps explain why our understanding of methodical ISD is blotted by conflicting research findings, disagreements about concepts, contradictory theories and glaring gaps in our body of knowledge concerning this activity. We69 have conflicting evidence that ISD methods are ever used, or work successfully if they are ever used (Wynkoop and Russo 1993). Much of the existing research is missing the fundamental questions:
"In short, methodology knowledge is based on conceptual writings and studies of small systems in unrealistic contexts, augmented by surveys that often compare unspecified methodologies (e.g., comparing prototyping to SDLC use). . . We do not understand how methodologies are selected or adapted or how they should be selected — or if they should be selected at all. We don't know how methodologies are used or how effective they are." (Wynekoop and Russo 1993, p. 186)

**Privileged: ISD is a rational, determined, and goal-driven process**

The fundamental sequence of events in many ISD methods requires the determination of a set of goals prior to ISD analysis and implementation of actions to achieve these goals. There is a shared assumption in this privileged text about a three-stage rational sequence: (1) determine goals, (2) determine steps and events that lead to these goals, (3) follow the steps and generate the events. This assumption closely follows the other privileged assumptions like causal linearity, reductionism and universality, but it does take these ideas one step further by assuming that IS developers will share the espoused goals and faithfully adhere to the plan by exercising their rational powers.

Thus preidentified goals serve as targets and motives for reasoned, well-considered and explicitly chosen sets of activities. For example, structured methods, information engineering, and prototyping methods situate information systems planning in the context of organizational goals and strategies. The decision about which systems are to be built and the order in which they are to be built is made on the basis of rational criteria which most logically aid the organization in meeting its given and explicitly stated goals.

This process is reminiscent of the "normal" scientific ideals that have been, through university training, inculcated in many IS developers: (1) state the theory, (2) operationalize the hypothesized relationship between variables, (3) manipulate the independent variables to achieve the desired effects on the independent variables. IS practitioners replace scientific rigor with professional rigor. They acquire positive knowledge about their settings, and by rational adjustments to systems components, they exercise positive control over their social environment.

As professional techniques, ISD methods borrow epistemological assumptions from their scientific reference disciplines. Essentially they provide a structural framework for the acquisition of knowledge about the environment following the data collection procedures of the natural sciences (cf. Neurath 1939, Nagel 1961). These ISD methods are the paradigm within which professionals learn about their surroundings and project their control over these surroundings.

**Marginalized: ISD is negotiated, compromised and capricious**

The methodical view makes rather different assumptions. First, the outset of an ISD project will find little agreement about a set of goals for the process of changing the system. In order to reach any espoused agreement the goals must be broad, unspecific, and ambiguous. The mechanism of ambiguity is used to permit human actors to interpret these vague goals differently. This tacit disagreement of individual interpretation leads to an ISD reality characterized by many conflicting sets of goals. It is extremely difficult to apprehend these conflicts for two reasons. First, the individuals operate within a dichotomy of theory-in-action contrasted with espoused-theory (Argyris and Schön 1978) which may prevent them from understanding and pronouncing their own set of goals in a redetermined way. That is, individual goals will be discovered as the ISD unfolds. A realistic set of unambiguous, achievable goals cannot be predetermined.

A second assumption holds that organizational reality is socially defined. The IS developers' knowledge about their environment is influenced by habitualized behavior,
institutionalized values politically reified observations, etc. (Berger and Luckmann 1966). The epistemological basis of IS developers' knowledge about their environment is interpretive rather than positive. The organizational reality is subject to revisions as ISD unfolds. These revisions may even be violently twisted, giving the project a "through-the-looking-glass" setting.

Third, the human actors in the ISD process do not always behave predictably and rationally. They are sometimes spiteful, playful, stubborn, capricious, eccentric, etc. Some people involved in the ISD process will be creative and ambitious while others will be unimaginative and removed. This behavior frames the social context in the ISD developers will negotiate the ISD project through the innately conflicting sets of goals. The negotiated aspect of the ISD outcome is significant because it may very well tie the final system characteristics to a very different set of purposes and actors than those that may have ostensibly predetermined an expected outcome. Individual charisma, persuasiveness, political power or tenacity may become a paramount influence.

Rather than being a rational, purposive, and goal-driven process, ISD is subject to human whims, talents, and the personal goals of the actors involved. From the amethodical view, ISD is a process by which some questionable observations are used to construct a set of ambiguous goals and disputable steps that are handed over to a group of independently-minded sprites who will proceed to do whatever they want to anyway.

This marginalized text seems to explain some observations about ISD that are problematic for the privileged assumptions. For example, Brooks (1987) acknowledges the shortcomings of ISD methods because they cannot displace human inspiration and creativity; they cannot "inflame or inspire the drudge" (p. 19). For Brooks, the important component in ISD is a great designer, a talented human being. Lee (1991) recognized that a preoccupation with positivist science as a criteria for our reference disciplines has led to ISD methods which ignore human meanings and values as fundamental elements of systems designs. He points to the merits of non-methodical, "creative" professions, such as architecture, as more appropriate reference disciplines for the field of ISD. Beath and Orlikowski (1994) deconstruct the relationship between the IS professional and the user in the context of the information engineering method. They found an ideology of user involvement that contradicted a deeply embedded dichotomy between users and analysts that reinforced an ambivalence to user concerns. This dichotomy is itself privileged, with the analysts dominant and the users submissive. This dichotomy conflicts with the user responsibility for ultimately using the system. Because this dichotomy might be shared by most ISD methods, the issue has practical implications for ISD in general:

"We believe the findings discussed in this paper raise a central challenge for the field of IS. In general, both the research and practice of systems development have tended to take for granted rather than question the structural distribution of power, authority, knowledge, control and resources that constitute the institutional context of systems development." (Beath and Orlikowski 1994, p.375)

These marginalized assumptions also help explain why the exact same design problem can engender completely different design solutions. For example, Turner (1987) studied the implications of wide variance in design solutions created by student designer teams. He suggested that many ISD methods disallow important design processes such as the "creative leap" by which conflicting choices are elegantly reconciled by an inspirational merger of opposing objective analyses. Peter Naur (1993) studied the use of ISD methods from the perspective of their suitability for the variety of human beings that may be drawn into the systems professions. Naur found that designers will fundamentally disagree on how ISD methods were intended to be practiced. He concludes that no method can hope to find universal practice because every method must be inherently unsuitable to some large portion of the population of professionals who are
charged with systems design.

The amethodical view suggests that each development project must actually include activity (however informal) that defines its own unique approach to solving each development problem. The amethodical assumptions suggest that ISD is a pastiche of activities, events and products. Some of these elements may be borrowed from various methods (including scientific methods or ISD methods), some elements borrowed from elsewhere (like art, literature, or architecture), and some elements invented altogether. Each element of the ISD pastiche is very likely to violate in some way the structures of any particular ISD method. Like an emergent grammar, ISD is a collection of systems idioms, proverbs, cliches, formulas, jargon, favored technologies, habitualized interfaces, typical notation, and so on. ISD is pasted together "on-the-fly", in a manner that is interactive and negotiated with the organizational actors and the rest of the environment of the ISD project.

Implications for the Goals of ISD

We admittedly bifurcate the concepts of method and amethod into opposing texts in each of the preceding sections. In one sense, this bifurcation did not raise a marginalized text from a privileged one, but instead created a second privileged text. The newly privileged text of amethodical was in effect presented as more important than the methodical counterpart. Deferral of meaning accepts that neither of these texts is the final word. The marginalized positions that we present are themselves imbued with cultural and intellectual assumptions and should be subject to a further deconstruction. The goal of deferral is not to displace one view with the other, nor is it to provide a kind of synthetic combination or compromise of the various views. Rather, one goal is to hold both views in mind simultaneously.

The danger in simply raising the marginalized views as a replacement for the privileged view might be thought of as the "costs" of the marginalized view. For example, the marginalized view makes various stakeholders' goals dominant over management's goals. The amethodical text raises the importance of systems developers listening to many voices, serving many masters and eschewing any single, clear purpose. The cost of this shift is that of clarity of purpose. In the privileged, mainstream view, management's goals were, for good or ill, clear, univocal and cheaper to fulfill from a transaction cost point of view.

There are other important examples of the costs of privileging the amethodical view. Embracing ambiguity and deferring the freezing of organizational and institutional structures surrenders the precision of the expected ISD outcomes and requirements definition. The amethodical view supports conflict over consensus. This view requires developers to attend different voices and interests. Political bargaining and negotiation lead to compromise solutions rather than assuming a consensus view of organizational settings. The cost is in the overhead of maintaining procedures for more-or-less constant negotiation and adjustment of ambiguous goals to changeable political decisions.

The amethodical view appreciates innovation and "organizational shake-ups" that lead to adaptation, experimentation and in turn to accidents and opportunism. There is a cost in allowing trial and error, and in adapting to accidents. The resultant solutions may just as likely represent a misfit as a good fit. The mainstream view favors calculated organizational fits and matches such that the system mimics the expected characteristics of the organization.

By privileging the amethodical we ignore method's nature as an ISD "coping" mechanism. Method provides a means for developers to cope with the essential fluidity of the social organization by ignoring continuous change and to freezing a particular organizational view in time. By privileging the amethodical we ignore the familiar and pervasive aspects of method that guide and balance developers, even those who might be unknowingly engaged in amethodical development. Developers cope through the use of method as a parable rather than a procedure.
Methods guide rather than direct the developer's process. Method is a means to frame the domain of inquiry and establish an initial conceptual and intellectual boundary on the problem. Privileging only the amethodical ignores the way that experienced developers intuitively approach the development process as a professional language game. In these particular language games the initial rules (or professional grammars) include the ISD method itself. Certainly the game and its rules evolve in use. This evolution regards the engagement between developers and their methods, and defines part of their professional discourse. This discourse describes the negotiation process through which the rules (method) are changed during development process. The amethodical view marginalizes the way this language game commences (e.g., the original set of rules or the method) in favor of privileging the way this game evolves.

By privileging the amethodical we lose the important role of methods as design metaphors that is often made by those that practice ISD (Coyne 1995). In creating their structured systems approach Yourdon and DeMarco were, after all, simply distilling their examples of best practice. Books presenting methodical approaches typically offer exemplars of successful ISD practice. Like good poems that work as distilled symbolic representation of some aspect of the human condition, these exemplars are simply distilled metaphors for the ideal development effort.

By privileging the notion of organizational emergence we marginalize those aspects of organizational life which are regular and appear to have structure. Because humans engage in social organizing activities, there must be a tacit agreement to engage in socially constructive activities that, in turn, leads to organizational regularities. The amethodical marginalizes social agreements that persist long enough to become organizational features in need of ISD modeling.

By privileging the amethodical we also marginalize the history of successful methodical systems development in large-scale projects. Large-scale systems involve hundreds of developers and multiple development teams. These projects require, at minimum, an agreement on which organizational language (e.g. method) to speak as a starting point to the project discourse.

Each of these counter-amethodical positions are themselves becoming privileged texts that involve the marginalization of other positions. This texts involve costs and invite further deconstruction. For example, recognizing that methods provide the starting point for the rules in a language game marginalizes the evolution of method. Once this initial rule-set consisted of structured methods, for many organizations this initial rule set has evolved into object-oriented methods. In this sense, ISD projects are continuous prototypes for the development of new methods.

**Conclusion**

Four questions were posed in the introduction to this article. We can draw insight into these issues from the deconstruction of the privileged method text. This insight also addresses the problems related to this text through important implications for research practice and education.

First we asked whether ISD methods really describe ongoing systems development practice and whether they explain why information systems are developed in certain ways. The strength of the amethodical reality suggests that we have a very limited understanding of how systems are developed. Methodical assumptions clearly define the paradigm of modern ISD thought so strongly that little research effort has been brought to bear on measuring amethodical phenomena. Notwithstanding the efforts directed at learning how to examine and analyze ISD processes (Sabherwal and Robey 1993, 1995, Newman and Robey 1992, Robey and Newman 1996) it is doubtful that our instruments are presently capable of measuring such phenomena if the state of the disciplinary discourse does not admit them as possibilities. Einstein once described the confinements of such strong theoretical assumptions: "It is the theory that decides what we can observe" (Heisenberg 1971). Methodical ISD so dominates our thinking that it has become a self-confirming construct, impossible to disprove in modern terms.
The marginalized text suggests that ISD unfolds differently than we previously believed and that developers adapt ISD methods to particular situations. Developers are successfully mixing and matching elements from seemingly contradictory ISD methods, e.g., structured prototyping (cf. Connell & Shafer 1989). We now have open research questions about these amethodical phenomena and the criteria for a successful amethodical ISD project. These implications include the very real need to develop instruments and research approaches that permit us to measure the degree to which ISD is methodical and amethodical.

We also asked whether methods actually function as frameworks, formulas or templates of successful ISD. This question raises the implications of the amethodical text for practice. Our improved understanding of the contrasting ISD texts legitimizes idiographic, one-shot development schemes that arise "on-the-fly" as a negotiation between ISD actors and the ISD environment. That is, this understanding permits ISD project managers to relieve system developers of the need to "pretend" that an ISD method is followed absolutely. Project managers are free to be more aware of how the system is actually being constructed. These managers would be encouraged to focus on the control of, and the value in, the dissimilarities (the pastiche) of the actual development activities. If ISD method becomes an obvious fiction in an ISD project, ISD managers could learn to recognize and benefit from the adaptive flux of the amethodical aspect of the process.

In addition, the amethodical text suggests a dominance of notation over process in the selection of an ISD method. ISD methods often embrace a notation set as well as a process for developing the notation. Possibly the adoption of the notation is more critical for defining how ISD will unfold, since the notation defines what can and cannot be represented by the analysis and design. This suggests that practitioners faced with decisions regarding ISD method adoption give strongest consideration to the notation prescribed by the ISD method, rather than the prescribed procedures or heuristics. Indeed, the most practical notation set might well be pasted together from various sources.

Finally, we asked if methods were unattainable ideals and hypothetical "straw men" that provide normative guidance to utopian development situations? Here we find educational implications, since the presentation of ISD methods also can be informed by the amethodical presence. ISD methods seem more like idealizations than prescriptions, and might better be presented as "cases" or "exemplars" rather than practical frameworks. This shift reveals the need to present a set of sound examples of how parts of various ISD methods can be mixed and matched (perhaps with other, newly invented parts); plus examples of how ISD approaches can be assembled "on-the-fly" by cannibalizing the ideals from the textbooks.

A further need in our ISD classrooms is for the concepts and criteria for innovating one-shot ISD approaches and negotiating the ISD activities with the ISD environment. Because of our methodical paradigm, we don't actually know how this happens "in the wild". Students need guidelines for recognizing elements of the ideal settings for discrete parts of ISD methods. For example, how and when should IS developers separate the notation from the procedure?

The answer to the questions raised in the introduction do not reside in definitive solutions or prescriptions for action. Rather in the fact of the asking and answering they become part of the endless chain of signification, the différence and the deferral that keep these notions at play. By opening questions about systems development without method, we may best clarify what it means to use a method in developing an information systems.

By adopting a single domineering concept of method all of our thinking about information systems development becomes imprisoned by this one concept. The method is not only our way of thinking about systems development, it is our way of thinking about "thinking about systems development." Just as there is an intellectual cost incurred by privileging the marginal (amethod), we must remember that the price of one domineering concept (method) is a tightly defined boundary for our knowledge about certain activities. In ISD, this price includes the perceptions
that we forego by assuming an unwavering methodical viewpoint. When the idea of method frames all of our perceptions about systems development, then it becomes very difficult to grasp its non-methodical aspects. It limits not only our understanding, but our empirical observations of how the development of information systems will unfold in an human organization. In other words, our obsession with method can cause us to ignore activities that do not fit within a methodical frame.

Acknowledgement

The authors acknowledge and appreciate the strong influence on this paper of discussions with Heinz K. Klein. We also gratefully acknowledge the comments of Keld Bødker, Paul Cule, Gordon Everest, Bernie Glasson, Mark Keil, Finn Kensing, Jungwoo Lee, Kalle Lyytinen, Eph McLean, Dan Robey, Jon Turner, four anonymous reviewers and our ISR associate editor.

Appendix

For readers less familiar with postmodernism, there are several ideas that are assumed in the body of the paper. These ideas regard postmodern thinking as it relates to organizational settings and post-structural organizational thought. One aspect of postmodernism implies that important values and beliefs underpinning modern religion, philosophy, art and science have imposed limits on human thought and activity. Breaking free of these limits releases the discovery of unnoticed dimensions in our world. This emancipation not only involves the uprooting of fundamental modern assumptions, but the even more difficult task of finding and exposing these unquestioned beliefs. Such postmodern exploration is in process in organizational analysis (e.g., Cooper and Burrell 1988\textsuperscript{86}), organizational theory (e.g., Boje 1994, Berquist 1993\textsuperscript{87}), and in ISD methodology (e.g., Beath and Orlikowski 1994\textsuperscript{88}, Coyne 1995\textsuperscript{89}).

Another aspect arises in the decentering of the author of the text. This decentering means that the intentions of the reader become paramount. The author, like the subject, is contingent, temporary, situational and ephemeral. What an author meant to say is immaterial because it is never recoverable, is itself clad in chains of signification and is, anyway, the subject of interpretation involving other (the readers') chains of signification. "Chains of signification" is an important semiotic concept in the work of structural linguists such as Saussure and Piaget. But there is an important twist recognized by Derrida and adopted by nonstructural linguists such as Paul Hopper. This twist acknowledges that chains of signification are a type of infinite recursion, always at play and never returning to an initial point of reference or the "transcendental ideal." Literature, art and science depend on this concept. Should the ideal meaning of every utterance become known there would be no need for further discussion. Thus, one can never claim to have "captured" an author's intention.

The marginalized texts of amethodical development benefit from two fundamental conceptual undercurrents. The first is the social construction of reality (Berger and Luckmann 1966\textsuperscript{85}) and Giddens' structuration theory. Many organizational structures like "departments", "policies", and "managers" are only part of reality as long as organizational members and organizational theorist agree that these are part of reality. These structures are institutionalized in human behavior and then reified by the institutions. The implications are that, if reifying institutions (say, research universities) decided that these organizational structures were not real, then these structures would cease to be real.

Giddens' (1979\textsuperscript{91}, 1984\textsuperscript{92}) structuration theory mediates structural thinking with social constructivism. Giddens finds a duality in the structure concept, recognizing that structure is an
abstraction that has no existence independently of the human actors, yet somehow constrains and guides the actions of those human actors. Structuration theory represents a moderation or softening of the concept of structure in human organizations: "It is thus more appropriate to speak of social systems as exhibiting structural properties that are produced and reproduced through the interaction of human actors, rather than as having structures." (Orlikowski and Robey 1991 93, p. 147) The duality of structure in structuration theory suggests that human organizations recursively create, and then conform to abstract structures.

A second undercurrent in non-structural thinking about organizations is the concept of emergence (cf. Truex and Klein 941991). Unlike structuration theory, which compromises on a structural duality, emergence theory is a complete alternative to the structure concept in science, grammar, and organizations. Emergence is the notion that human activity is dominated by change: "destructure", not structure. Reality, knowledge, grammar, etc. are in motion: no "states" exist, just "flux". The emergence concept also suggests why structure seems to appear. Emergence points to a social need to structure, arising from language use and formation in discourse. Non structural linguists describes it is an attempt to achieve structure where:

...like speech itself, [grammar] must be viewed as a real-time, social phenomenon, and therefore is temporal; its structure is always deferred, always in a process but never arriving, and therefore emergent. (Hopper 921987, p. 141-142)

Emergence is not a sequence of succeeding structures, but a continual movement toward structure without every attaining a steady-state. Emergence is the postponement or deferral of structure, leaving grammars, organizations and science as provisional, negotiable, and epiphenomenal.

The denial of structure as a preexisting and required condition of grammars in linguistics supports a similar denial of structure as a preexisting and required condition in organizations, thus giving rise to the concept of emergent organizations. This emergent organizational concept regards adaptation and flexibility as a dominant feature of the organizational landscape which requires a ISD approaches which acknowledge and adapt to emergent organizational phenomena.

Thus, in the second instance or marginalized notion of the amethodical, we note that the organization upon whose functions or data meanings the information system is being modeled is a shadow of that which it is not. The organization may be described as a meaning creating system and one simultaneously engaged in the deferral of meaning. Similarly the system being developed may be viewed as one which defers meaning. This may be illustrated as follows. Systems are derived from symbolic representations of organizational function, processes or data. The IS is constructed from specifications which are themselves derived from requirements from a thing in transition. The IS is already a third or fourth order signification of the organizational element being modeled. There is a continuous chain of signification at each stage pointing backwards to a fluid and transient (e.g. emergent) phenomena, a social organization.

References


Boland, R. (1979) "Control, causality and information system requirements", Accounting, Organizations and Society 4 (4) 259-272.


Calmes, Françoise; Charbonnel, Gilles; and Dumas, Philippe. (1991) "Merise et Ossad: deu méthodologies à comparer in Autour et a L'Entour de Merise. April, Sophia Antipolis.


Eisenhardt, Kathleen M. (1989) "Agency Theory: An Assessment and Review", Academy of


Boland, R. (1979) "Control, causality and information system requirements", *Accounting, Organizations and Social* (4) 259-272.


