
Symbolism and Information Systems Development: Myth, Metaphor and Magic

Author(s): Rudy Hirschheim and Mike Newman

Source: *Information Systems Research*, Vol. 2, No. 1 (MARCH 1991), pp. 29-62

Published by: [INFORMS](#)

Stable URL: <http://www.jstor.org/stable/23010612>

Accessed: 27/04/2013 17:49

Your use of the JSTOR archive indicates your acceptance of the Terms & Conditions of Use, available at
<http://www.jstor.org/page/info/about/policies/terms.jsp>

JSTOR is a not-for-profit service that helps scholars, researchers, and students discover, use, and build upon a wide range of content in a trusted digital archive. We use information technology and tools to increase productivity and facilitate new forms of scholarship. For more information about JSTOR, please contact support@jstor.org.



INFORMS is collaborating with JSTOR to digitize, preserve and extend access to *Information Systems Research*.

<http://www.jstor.org>

Symbolism and Information Systems Development: Myth, Metaphor and Magic

Rudy Hirschheim

*College of Business Administration
University of Houston
Houston, TX 77204*

Mike Newman

*Department of Accounting and Finance
University of Manchester
Manchester, U.K.*

It is our intention to challenge the commonly held assumption that information systems development (ISD) can be conceived of as a normative process reflecting conventional economic rationality. We ask: is systems development the rational process so eloquently described in the “classic” works of DeMarco (1978), Gane and Sarson (1979), Weinberg (1980), Yourdon (1982), Jackson (1983), and Martin (1985)? Or does this orthodox view fail to explain the *actual* practice of systems development? It is our view that even the basic assumptions about the rationality of the actors and the social processes they engage in need to be critically appraised. We suggest that if the assumptions about economic rationality are closely analyzed, it can be seen that they do not reflect the reality of systems development. ISD tends to defy rational explanations. As an alternative to basing our understanding of systems development on economic rationality, we contend that symbolism holds more promise. Instead of focusing on data flow diagrams, structured walkthroughs, requirements specifications, and the like, we concentrate on the role of myth, metaphor, and magic. These concepts offer considerable scope in interpreting the social actions that are embodied within ISD. We feel they facilitate a much richer understanding of systems development.

Myth—Magic—Symbolism—Metaphor—Social aspects—IS development—Qualitative research methods

1. Introduction

Over the years, a great deal has been written about information systems development (ISD). It is a topic of growing interest as organizations increasingly recognize the role computer-based information systems (IS) are likely to play in their survivability. Indeed, the strategic use of new information technology and systems is seen as the key to the future (Ives and Learmonth 1984, Porter and Millar 1985, Parsons 1983, McFarlan 1984, Beath and Ives 1986). Yet the history of ISD does not paint a bright picture. The past decades have seen numerous failed systems—some of the monumental variety, others more mundane (Lucas 1975, Lyytinen and Hirschheim 1987). According to Mowshowitz (1976) many, if not most, information

1047-7047/91/0201/0029/\$01.25

Copyright © 1991, The Institute of Management Sciences

systems are failures in one sense or another. More worrisome, perhaps, is the survey done by Gladden (1982) who reports that in 75% of all the cases of systems development he surveyed, either the development is never completed or the product of completed development is not used. Subsequently, some commentators have gone so far as to refer to a "crisis in systems development" (Sibley 1986, Martin 1985, Bubenko 1986).

Given the strategic importance of information systems for both now and the future, it is imperative that we understand why so many ISD efforts have not been successful and why ISD has been so problematic. In the literature one can, of course, find as many reasons for ISD failures as the numbers of failures themselves (Lyytinen and Hirschheim 1987). It is our contention that there is one underlying reason systems development is as problematic as it is, and why there have been so many failures: ISD has been too narrowly conceived. A major part of the problem has been the inadequate recognition that ISD is largely a social process (Keen 1981, Checkland 1981, Mumford 1983, Markus 1984, Kling and Iacono 1984, Robey and Markus 1984, Newman and Rosenberg 1985, Lyytinen 1986, Hirschheim, Klein and Newman 1987). Treating systems development as largely a technical process is now seen by many designers—although not all—as a recipe for disaster.

It comes as no surprise to systems developers that social interaction is important: obtaining requirements, discussing design options, performing structured walkthroughs, prototyping, and the like, are all intensely social in nature. But while it might be recognized that ISD is essentially a social process, this in and of itself is not sufficient to guarantee success. For quite some time, conventional systems development approaches have acknowledged the importance of the social element of ISD. Nevertheless, they concentrate on the technical process of systems development. They equip the developer with neither the tools nor the knowledge for dealing with the social processes intrinsic to ISD. Simple platitudes such as "get the support of senior management" or "involve the end user" are hardly sufficient to guide systems development. They tend to mask the social nature of ISD or portray it in simplistic ways. They do not allow developers to understand, let alone fully appreciate, the social nature of systems development.

Developing an understanding of the social nature of ISD is, however, by no means a simple matter. It is first necessary to conceive of systems development as largely a social process which relies to an increasing extent on new information technology for its effective operation (Goldkuhl and Lyytinen 1982, Land and Hirschheim 1983). Systems development proceeds through the social interplay of multiple actors who attempt to interpret or "make sense" of their and others' actions, largely through the medium of language (Boland 1985). Attempts to construct "theories" of ISD which account for its social nature are beginning to emerge (Ciborra 1985, Boland and Day 1982, Markus 1984, Lyytinen 1986, Hirschheim, Klein and Newman 1987, and Kling and Scacchi 1982). They are all, unfortunately, at an elementary stage of development. Further work must be done to see how well they (a) represent or explain systems development and (b) aid developers in their tasks of designing, building and implementing information systems. In this context, we propose to make sense of the social nature of ISD by relying on the use of *symbolism*, i.e., the practice of representing ISD situations in terms of symbols.¹ We feel such an approach permits a

¹ Symbolism has a subtly different meaning for ISD participants and ISD researchers. For the former, symbolism connotes the expression of a set of attitudes and beliefs (including the creation of symbols

richer understanding of ISD, as we will proceed to show in the remainder of this paper.

The value of a symbolic approach lies in its ability to interpret seemingly irrational events as symbols which serve one or more actors' rational functions. Indeed, symbolism has been widely used in anthropology and sociology to help interpret the actions of social actors. It has more recently found its way into the organizational theory literature to explain the role and function of management (Morgan 1986, Trice and Beyer 1984, Feldman and March 1981, Gowler and Legge 1983, Kets de Vries and Miller 1984, Westerlund and Sjostrand 1979, Pondy et al. 1983); into the accounting literature to explain budgeting (Boland and Pondy 1985, Covaleski and Dirsmith 1983) and other phenomena (Gambling 1977); and even, to some degree, into the information systems literature (Lanzara 1983, Robey and Markus 1984, Boland 1987). Here, we draw on the work of those who have applied the symbolic approach to other domains to provide a new way of conceiving of and thinking about ISD. In order to do so, we adopt a phenomenological position from which we see the world as a "script." The approach allows us to interpret the world—in this case, ISD—in terms of symbols. We define a symbol as an image used for, or regarded as, representing something else. Symbols give meaning to what is perceived; they act as the filter through which the script "is read." This is a hermeneutic exercise and has a rich historical tradition, particularly in theology, where the "script" to be interpreted is the scriptures (cf. Gadamer 1976, Habermas 1984).

The paper proceeds as follows. In the next section we outline the notion of symbolism which we use as the mechanism for interpreting systems development. We specifically focus on three key symbols: myth, metaphor and magic. We also look at their role in helping us understand ISD, and explore examples of these three symbols in the literature. Section 3 provides some empirical evidence from case studies of myth, metaphor and magic. These examples show how symbolism manifests itself in practice. In particular, we show evidence to lend further support to some of the examples from the literature discussed in §2; for each of these three kinds of symbols, we also show evidence for examples not discussed in §2. Section 4 discusses why a symbolic interpretation of ISD might lead to better systems development in the future, specifically in terms of the movement toward multi-perspective approaches in development methodologies. Section 5 concludes the discussion by pointing out the value of symbolism, particularly as a vehicle for actors to simplify their world, as well as the potential dangers of ignoring symbolism—inappropriate designs, conflict, and dysfunctional behaviors.

2. Symbolism and Information Systems Development

Symbolism is not a new notion; anthropologists and other social scientists have recognized its importance for centuries. Philosophers such as Aristotle also recognized the value of symbolism, and were intrigued by the use of metaphors (Ortony 1979). Whitehead (1927) may have captured the essence of symbolism best when he wrote: "Symbolism is no mere idle fancy or corrupt degeneration: it is inherent in the very texture of human life" (quoted in Morgan et al. 1983, p. 3).

More recently, organizational theorists have been concerned with the role of symbolism in organizations, and specifically, the relationship between symbolism and

themselves) in order to make sense of their ISD situations. For the latter, symbolism is an analytical approach for understanding the behaviors of ISD participants.

organizational culture (Pondy et al. 1983, Frost et al. 1985). Organizational culture is often perceived as the reason that organizations (and more specifically, organizational behavior) are not generalizable (Van Maanen 1985). And it is precisely here that symbolism is thought to be helpful. Staw (1985) for example argues that symbolism may have greater “predictive power over more conventional observations of variables” (p. 117). His example of an organizational argument, which on the surface may be interpreted as a hostile action, but symbolically is a form of supreme compliment because the combatant acknowledges the value of his opponent’s argument by contesting it, suggests how symbolism provides a richer understanding of organizational behavior. Of course there are those such as Alvesson (1984) who question whether symbolism is relevant in studying organizations, because they are not like the well-integrated units of nations or tribes where symbolism is most prevalent. But according to Turner (1986), such a view is misguided: symbolism is an important weapon in the arsenal of the organizational scientist.

Organizational researchers such as Wexler (1983), Smircich (1983), Feldman (1989) and Dandridge (1983) see the value of symbolism in its fundamental ability to permit people to act. Astley (1984) writes:

Symbols . . . do not just fulfil expressive, sense-making functions for managers. . . . By fixating individual perception on common beliefs and values that de-emphasize differences between organizational participants they bridge across idiosyncratic cognitions to produce a basis for coherence that justifies and facilitates the mobilization of collective action. In this respect, theories, worldviews, goals, visions, expectations, plans, myths, stories, rituals and terminology affect practice not directly, but indirectly, through ideologies which fuse organizations into a unitary body and thereby lubricate the process of change (p. 270).

Mitroff (1983) echoes this thought when he states that symbolic meaning “provides the emotional comfort that is needed in coping with a precarious and often terrifying world” (p. 388).

Thus, symbols play a critical role in organizational life. They are an important means for simplifying and understanding a complex world. They help reduce uncertainty, facilitate interaction and communication, clarify relationships, and generally, make people feel more comfortable with their surroundings. Managers use symbolism all the time; it helps them to interpret the significance of their actions and the actions of others.

According to Dandridge (1984), symbols—particularly ceremonies—are concerned with social maintenance; they offer actors the opportunity to practice organizationally useful behavior in the arena of the social community. Eoyang (1983) sees the value of symbolism in its ability to facilitate the communication of meanings between individuals. Symbols are a powerful vehicle for conveying deep-rooted meanings. They are critical for organizational sense-making. Indeed, if symbols are stripped away from the societies in which they are embedded, “all sorts of psychological and social confusion occurs. When the ‘life supporting illusions’ are destroyed, communities go to pieces” (Bowles 1989, p. 410).

Traditionally, symbols evoke a negative connotation—that myths, for example, are opposed to reality, that they are antithetical to fact—but they need not. As Boland and Hoffman’s (1983) analysis of symbols in a machine shop has shown, symbols do have a positive value. The specific symbol they analyzed, humor, was shown to convey important patterns of meanings which helped the workers to frame and control their jobs while allowing them to reaffirm their identities within the work group. Similarly, Smircich (1983), in her case study of an insurance company, showed how symbols

such as rituals provided a common ground for organized action. It must be noted, however, that such positive interpretations of symbols are in the minority. In an analysis of the management literature, Ingersoll and Adams (1986) report that symbolism is treated as a negative phenomenon in 80 out of the 85 papers they looked at. So while symbolism can be seen in a positive light, in the management literature it traditionally is not.

Morgan et al. (1983) define a symbol as a sign which “denotes something greater than itself” (p. 4). The key feature of symbols is that they are “created subjectively and are invested with a particular kind of subjective meaning” (p. 4). They are a means of expressing and sustaining valued patterns of beliefs. According to Morgan et al. (1983), symbolism is the practice of symbolic representation. In this paper, we define symbolism as the practice of representing ISD situations in terms of symbols. It is the expression of a set of attitudes and beliefs. Here, we view our role as one of documenting the behavior of ISD participants who are trying to make sense of their ISD situations, and classifying the symbols they use for sense making. We classify these into three types: *myths*, *metaphors* and *magic*. There is no set of universally accepted categories of symbols in organizations. The “myth, metaphor, and magic” triad accounts for most of the phenomena we observed in our case studies. Also, many of the notions in the literature can be accommodated. For example, ceremonies or rituals can often be seen as “magic.” We do not enter into the debate on the construction and propagation of symbols—how people learn to use symbols, how they are formed, how they are communicated and perpetuated (Tinker 1986). Neither do we consciously attempt to privilege some symbols over others.²

Symbolism is, of course, largely unobservable directly. It is by observing the behaviors of systems designers that we infer the attitudes and beliefs they hold. The process of attitude and belief formation (socialization processes, word-of-mouth anecdotes, training, etc.) is highly important and has received some consideration in the IS literature (Kumar and Welke 1984, Vitalari and Dickson 1983, Hedberg and Mumford 1975, Dagwell and Weber 1983, Mumford 1972). For the purpose of this paper, however, we will consider only the actual behaviors themselves when observing the people involved in ISD. The evidence for the behaviors comes from case studies which are described below. The relationship between behaviors and symbolism is complex, involving an understanding of attitude/belief formation in communities, and the adoption of those beliefs/attitudes by individuals. Symbolism first and foremost revolves around shared meanings: patterns of beliefs, rituals and myths, which evolve through time and function as social glue, binding communities together (Smircich 1983). By observing the behaviors of subjects, and by a careful interpretation of the records of their behaviors gathered through the case studies, we believe it is possible to infer features of the subjects’ attitudes and beliefs (i.e., the symbols they use). Consequently, we believe that the filters or symbols used by the subjects to “read” the world can be explicated.

Here we concentrate on interpreting ISD in terms of three key symbols: myths, metaphors, and magic. These symbols are usually inextricably linked during the processes of ISD, but we shall try to deal with them individually.

² It is clear that the researcher and the research process are themselves influenced by a symbolic interpretation of the world. We are in a real sense “trapped” in our mental models which form an “iron cage” around us, permitting us to observe certain artifacts while denying us access to others.

2.1 *Myth*

Myth can be defined as “A dramatic narrative of imagined events, usually used to explain origins or transformations of something. Also, an unquestioned belief about the practical benefits of certain techniques and behaviors that is not supported by demonstrated facts” (Trice and Beyer 1984, p. 655).

Myths are important for a number of reasons. According to Bolman and Deal (1984) they provide explanations, reconcile contradictions, and help resolve dilemmas. Myths can serve a number of functions: (1) myths explain, (2) myths express, (3) myths maintain solidarity and cohesion, (4) myths legitimize, (5) myths communicate unconscious wishes and conflicts, (6) myths mediate contradictions, and (7) myths provide narrative to anchor the present to the past (Cohen 1969). But myths are also ambiguous. According to Bolman and Deal (1984) they can distort our images and misdirect attention, but “despite their negative traits myths are necessary to create meaning, solidarity and certainty—myths keep us sane.” Myths transcend ordinary rules and logic. Pondy et al. (1983) state “anomaly and contradictions can be resolved with mythical explanations” (p. 163).

Myths tend to be communicated to insiders and outsiders via the medium of the story. “A good story provides a pleasant way of responding to unpleasant facts, they help to address problems of morale, security and legitimacy” (Bolman and Deal 1984, p. 158).

Bowles (1989) sees myth as an important tool in the analysis of organizational practices. We have identified six myths that systems developers use as guides to design, the first five of which have been investigated by Newman (1989). They will be explored below in turn, using evidence from the literature, and in §3.1, using empirical evidence produced by Newman (1989) to show the dangers of following them slavishly.

1. *User Involvement Is Beneficial and Should Be Encouraged.* This myth stems from the widely-held belief that users should be encouraged to participate in designing systems which will affect them personally. Manifestation of this myth may vary from the traditional rationality where the designer extracts facts from the users and prepares the design independently, to an alternative rationality in which the design team is headed by the user and involvement is continual. However, regardless of the approach, the designer may still interpret involvement very narrowly, not allowing users to radically challenge the proposed design.

That user involvement is beneficial is a widely-held belief, and almost every analyst will advocate its practice in ISD. Its benefits are extensively discussed in the literature (e.g., Bjorn-Andersen and Hedberg 1978, Mumford 1983). For example, in order to obtain user requirements for the new system, the key users must be involved. User involvement is also supposed to defuse resistance. Furthermore, it is promoted because users should have the right to influence the system they are going to use. Building and allowing users to experiment with mock-up systems to test out designs (prototyping) is the latest variation of this issue. While user involvement can be beneficial, a number of writers suggest that its practice may raise more problems than it solves (cf. Ives and Olson 1984, Opperman 1986, Keen et al. 1982, Robey and Markus 1984). For example, user involvement can be subject to manipulation by the designer, making it more of a weapon than an aid; furthermore, it can easily precipitate conflict among the user groups.

2. *Resistance to Systems Development Is Dysfunctional and Should Be Eradicated.* This myth starts with the belief that resistance to ISD is almost inevitable in large scale systems development. Markus (1983) studied a divisionalized organization where the corporate accountants attempted to wrest control over financial information from their divisional counterparts. When the divisional accountants realized what was happening, they resisted any further erosion of their control. The new system was centralized, a design that was contrary to the decentralized structure in the organization. In other words the system and the organization did not, and indeed could not, fit together. As Markus concludes:

The analyst should recognize that the goal of the exercise is not to “overcome” resistance, but to avoid it, if possible, and to confront it constructively if not. In some cases, this indicates that the implementor may have to lose the battle and sacrifice a pet system project in order to win the war. Resistance is not a problem to be solved so a system can be installed as intended: it is a useful clue to what went wrong and how the situation can be righted (p. 441).

3. *Information Systems Should Be Integrated Wherever Possible.* This myth is based on the belief that organizations are full of dysfunctions and that management information systems (MIS) can eliminate them by integrating functions from the different units. By doing so, managers can apply more rational approaches to management (Argyris 1970). This belief goes back to the time in the 1960’s and early 1970’s when the total systems concept was debated in the literature (e.g., Dearden 1966). The same utopianism now finds expression in applying technologies such as data base management systems, and the concept of Information Resource Management (Synnott and Gruber 1981).

The integration motive offers many advantages, but it is also a source of much conflict within organizations, as it undermines the existing commitments and power structures:

MIS undercuts the existing “rules of the game” as these are tied to the variety of weak or strong political systems in organizations which unify the various layers of management. Such a supposedly neutral act as one manager supplying data from his division to a common database . . . finds . . . that the data becomes the property of all units in the organization, increasing the potential of senior managers to control middle level managers directly (Newman and Rosenberg 1985, p. 403).

4. *The Systems Developer Is Generally the Best Person for Making Decisions About the System.* Despite the many failures and unfulfilled promises reported in the ISD literature, the zeal of many systems analysts remains surprisingly strong. The culture of systems staff inculcates a belief in its own mission to civilize organizations, producing a “we know best” attitude (Rose 1969, Franz and Robey 1984). Armed with an arsenal of technical wonders, the analyst is often only too willing to make policy decisions when managers abdicate their responsibility:

What tended to happen in the absence of definition was that the people responsible for systems analysis made policy decisions about the various controlling factors almost without being aware of doing so. There was therefore the danger of long-term decisions being made in terms of their appropriateness to systems design rather than in relation to more fundamental criteria. Thus real control tended to shift to the systems designers. (Hedley 1970)

The literature on ISD is replete with evidence about the differences between users and systems staff, and the misunderstanding these differences cause (Rose 1969,

Hedberg and Mumford 1975, Newman and Rosenberg 1985, Hirschheim, Klein and Newman 1987). Rose, for example, contrasts the position and goals of the systems personnel with those of the users:

Conflict is structured into their relationships. . . . Resentment of the innovation is aggravated by resentment of the innovator: resentment of his rewards, his values, his manner, his vocabulary, his prestige as change-maker, his apparent freedom from "normal company discipline," his promotability, his dress, his fickle loyalty to the firm, and, not least, his youth (Rose 1969, pp. 169–170).

This is echoed by the findings of Mumford (1972) who also found a wide gulf between the designers and the clerical staff in attitude and behavior: "They [the systems designers] don't give us much information. They don't tell us what is going to happen. It just happens and we are told about it." . . . "They just give you a book to read. There is little explanation of the new things that are happening." . . . "The user cannot resist, he can only be reluctant."

The belief or myth of "we know best," while more muted than in the late 60's and 70's, still, we believe, remains a powerful force which shapes ISD.

5. *Politics Should Not Be the Concern of the System Developer.* This commonly-believed myth is also related to the view that organizational dysfunctions can be eliminated by introducing well-designed systems. Politics is often viewed by many designers as part of the problem of organizations, as if it were an irrational aberration which needs to be eliminated or curtailed. While designers can to a large extent insulate themselves from politics when developing small systems in single departments, no such protection is available when major systems which cross organizational boundaries are introduced. If designers disdain politics or see their role as separate from it, they will never be able to play an effective part in the negotiation process vital to introducing such systems. In these circumstances, ISD and politics are inextricably entwined.

For her case study, Markus (1983) specifically acknowledged the role of power and politics in the design of the Financial Information System (FIS). Her view is that the FIS was a deliberate strategy of the corporate accountants to wrest more control from the divisional accountants:

The way in which FIS was designed implied a major gain of power for corporate accountants relative to their prior position *vis a vis* the divisional accountants. Prior to FIS, divisional accountants summarized raw data on the transactions in their divisions and sent the summaries to the corporate accountants for consolidations. . . . After FIS, however, all financial transactions were collected into a single database under the control of corporate accountants (p. 438).

Whereas the users in these cases seemed only too aware of the political society in organizations, it was rationality that was constantly stressed by the systems analysts. Mumford (1972) contrasts the approach of systems staff in designing the technical features with the non-technical, human element:

Human relations and the ability to manage the human part of change are skill areas which are dealt with intuitively and unsystematically and this is in striking contrast to the technical side of computer systems where the programmers and systems analysts continually stress the need for a logical approach.

6. *The Key to Successful Design Is the Use of a Top-Down Approach.* A sixth myth not often heralded in the literature anymore but nonetheless visible is the belief in the "top-down approach." To some systems staff, the top-down approach to systems

development offers a mythical quality to the design of systems. This has a history dating back to the 1960's, when the "Total Systems" approach was extensively debated in the literature (cf. Dearden 1966, Young 1968). While the "total systems" approach has fallen into disrepute because of its failure to produce workable systems, many systems people still cling to the top-down approach as a kind of touchstone or ideal form.

There are other well-known (and not so well-known) myths which can be found in the IS literature; for example, "the mythical man-month" (Brooks 1975), "the total systems concept" (Dearden 1966), the "seven mortal sins of systems work" (Lyytinen and Lehtinen 1987), the myth of the "true and complete specification of requirements" (McMenamin and Palmer 1984), the myth that structured analysis mitigates the disruption of organizational politics because it "makes analysis procedures more formal" (DeMarco 1978), the myth that "system integration is a virtue" (Kraemer and King 1979), the myth that software accounts for the major (and growing) proportion of an organization's IS budget (Ein-Dor 1988), and the myth that we can find a process that allows software to be designed in a perfectly rational way (Parnas and Clements 1985).

2.2 Metaphor

Metaphors are commonly-used vehicles for understanding. The use of metaphor is "a way of thinking and a way of seeing that pervade how we understand our world generally" (Morgan 1986, p. 12). A metaphor is essentially a way of understanding and experiencing one kind of thing in terms of another. According to Lakoff and Johnson (1980):

Metaphor is pervasive in everyday life, not just ordinary language but in thought and action. Our ordinary conceptual system, in terms of which we both think and act, is fundamentally metaphorical in nature. . . . The way we think, what we experience, and what we do every day is very much a matter of metaphor (p. 3).

Morgan (1980) concurs. He states:

Metaphors are not to be seen as representations of a reality "out there," but as tools for capturing and dealing with what is perceived to be "out there." . . . The process of metaphorical conception is a basic mode of symbolism, central to the way in which humans forge their experience and knowledge of the world they live in. . . . More fundamentally it is a creative form which produces its effect through a crossing of images (p. 610).

According to Pondy, metaphors have a dual purpose—that of facilitating change while maintaining stability: "In organizing, the use of metaphor *simultaneously* facilitates change *and* reinforces traditional values . . . metaphor can fulfil the dual function of enabling change and preserving continuity" (Pondy 1983, p. 164).

However, although metaphors are pervasive and sometimes helpful, they can also mislead. Boland (1987) identifies five metaphors which he feels guide systems development, but which are really dangerous fantasies "and not suited for guiding serious thought" (p. 367). They reify the human actor in ISD, and inevitably lead to dysfunctional consequences.

1. *Information Is Structured Data*. This most basic metaphor sees information as an object or entity. This metaphor is embraced in order to get around the hermeneutic problem of interpreting information. Instead of being concerned with the *meaning* of information to a free, intentional human actor, this metaphor treats

information as some publicly observable, immutable object. The metaphor imposes a stereotypical, rationalistic version of meaning, values, and significance on the actors (Boland 1979).

2. *Organization Is Information.* This metaphor permits an image of organizational control as the differential distribution of information and decision parameters. Organizations can be guided and controlled by the manipulation of structured data. This metaphor emphasizes a highly “rationalistic way of characterizing organizational life as goal-driven and purposive which further legitimates ignoring the individual actor’s need to interpret and make sense of organizational situations.” (Boland 1987, p. 369)

3. *Information Is Power.* This metaphor creates the image that information is power, in that it permits control over an individual. Power is conceived of as a one-way relationship. This metaphor transforms information and power into entities which are manipulated. The role of the human actor is further removed. The metaphor inflates the role of system developer to that of one who, through system design, creates and reallocates power. Yet it neglects the duality of power relationships. As Boland (1987) notes, power follows from the dialogue of organizational actors who attempt to interpret and understand their organizational world.

4. *Information Is Intelligence.* This metaphor links information to human intelligence. Information is seen as that which allows us to move through a problem space, and it is seen as equal to intelligence. Here, intelligence is reified; it is transformed into an object which is contained and localized in a computer. Boland (1987) suggests that the “result is the complete removal of human beings and their problems of action and sense making from the domain of information systems discourse” (p. 370).

5. *Information Is Perfectible.* This last metaphor is the culmination of the other four. Here, information is fantasized as capable of being perfect and “true.” Systems can be developed to supply the decision maker with “perfect information.” This metaphor removes the concrete historical moment of the situation from consideration, and discusses systems design in terms of a timeless, context-free, ideal future. But for this to be possible, we need “complete and error-free knowledge.” This is something which Boland (1987) argues is impossible. Citing the examples of John Kennedy’s assassination and the oil crisis of 1973, he contends that situations “will always be open to interpretation and reinterpretation. . . . The world has no single, immutable meaning, which the possibility of perfect information requires.” (Boland 1987, p. 375)

Hirschheim (1986) uses the notion of metaphor to describe the relationship between technology and its users (man). (In the context of ISD, technology can be interpreted as an information system.) Hirschheim uses two dual metaphors to discuss optimistic and pessimistic scenarios for the man-technology relationship.

1. *Technology as tool and man as craftsman.* In an optimistic scenario, the metaphors “technology as a tool” and “man as craftsman” are used. Technology is thought to be a tool in the hands of the workers. It is used when and where appropriate, to make their work more efficient and to raise the quality of life in general. The tool is of itself neutral, and can be used in many ways. Man is looked upon as a craftsman who scans his surroundings, choosing the most appropriate tools for the

task at hand. The craftsman is skilled and can apply the tools to advantage. Should the tools be unsatisfactory, the craftsman can modify them or choose not to use them. Through time, the tasks that the craftsman performs are likely to become more sophisticated; he then comes to rely on new and better tools for help. The development of new tools often drives the development of new skills and crafts. The relationship between craftsman and tool is that of master and slave.

2. *Technology as governor and man as machine.* In a pessimistic scenario, the metaphors “technology as governor, man as machine” are used. Here, technology is used to govern or control the operation of some task(s). For the governor to work properly, the tasks have to be highly structured so that all possible variances can be responded to. The governor is applied to a machine which performs activities in a structured and routinized way. Machines are built (trained) to engage in certain tasks and are provided with the appropriate raw materials. When machines wear out, they are discarded and usually replaced by more modern and better ones. Machines are capable of only action, not thought. They cannot suggest improvements in the types of activities to be performed or how they are carried out. They are under the control of the governor which regulates what they can and cannot do. The governor is not neutral, in that it is set by those who are ultimately in control. The relationship between machine and governor is also master-slave, except that in this case it is technology which is the master while man is relegated to the slave.

The Battle Metaphor. The military metaphor of “battle” is particularly powerful, and is frequently evoked by both users and developers in systems development. Keen (1981), for example, portrayed systems implementation in terms of battles played out between users and developers. Each would adopt particular offensive and defensive strategies to overcome the other party. This would be predicted by Lakoff and Johnson (1980) who note the pervasiveness of this metaphor in everyday life. They showed the importance of “battle” in a wide variety of interactions between both individuals and groups. Raspa (1984) calls this “a stunning metaphor for evoking an atmosphere where one’s survival is constantly at stake” (p. 10).

Other examples of metaphors can be found in the IS literature; for example, “information systems as competitive weapons” (Ives and Learmonth 1984, Parsons 1983, Wiseman 1985), and “information systems development as engineering” (Land 1989, Buckingham and Land 1987). See also Madsen (1989) who used the metaphors “warehouse” “store,” and “meeting place” to describe a library for which a system was being developed. As will be seen in §3.2, our case study data provides further support for the “man as machine” metaphor, as well as for one additional metaphor, namely “organizations as fiefdoms.”

2.3 Magic

Magic has played, and continues to play, an important organizational function. Like rituals, it binds people together; it is a mechanism for group solidarity. Earl (1983) defines magic as “The superstitious or religious, as opposed to the scientific method used to control nature for a definite practical end, particularly to aid the functioning, binding, and survival of a society” (p. 128).

Of course scientism, a belief in the infallibility of the scientific method to reveal knowledge, is itself a form of magic. Cleverly (1971) defines magic as “Beliefs that cannot be destroyed by the presentation of contrary evidence, and the practices whose continuance is independent of their efficacy” (p. 9).

Although magic may be “unscientific” and “irrational”, Cleverly (1971) makes the following point:

I am not advocating any attempt to root out all magical and religious behavior from management. To do so would be to disorient the entire [management] culture and invite disaster. Moreover, it would be impossible. . . . To disturb that would be dangerous. To understand it is essential. For if we wish to understand the people we deal with, to influence them and control them—to manage them—the worst mistake we can make is to assume that the manager, even in the twentieth century, is a rational being (p. 12).

However, because management is expected to be objective and rational, these myths and rituals become taboo—they cannot be discussed openly. (Cleverly 1971, p. 150)

Cleverly (1971), in describing magic in more detail, identifies two dimensions associated with human behavior: both are dichotomous. The first dimension reflects the purpose of human behavior—the “ends.” Behavior is either “instrumental” in that it is directed at controlling or influencing the environment, including other actors; or it is “expressive” in that it is directed at the releasing of inner conflict or emotion. The second dimension reflects the nature of how beliefs are held. Cleverly distinguishes between the “skeptics” and the “believers.” The former do not believe in, or adopt, any theories or practices unless they can see them working and producing positive (and better) results. For the skeptic, truth is a transient phenomenon. “Believers,” on the other hand, are those who have steady faith in what they believe in. They stick firmly to their beliefs even when confronted with disconfirmatory evidence, often to the extent of denying their own sense perceptions because truth is construed as eternal and unquestionable.

In this paper, we distinguish magic from myth. Whereas the image created by myth has some basis and anchor in our collective experience (“reality”), the image created by magic, while appearing to be “real,” is but a facade (cf. a hologram). Magic manifests itself in ISD in many areas.

1. *The Systems Developer as High Priest.* The image of the systems developer is not uncommonly portrayed as that of the “high priest of technology,” the individual (or group of individuals) who can harness the power of computer technology to the benefit of the organization. The developer in such a role uses a variety of rituals and ceremonies which add to the imagery of the “high priest.” Good examples of these rituals are the “structured walkthrough,” the “sign-off,” and the “system test” (DeMarco 1978, Gane and Sarson 1979, Yourdon 1982). Such rituals are important to the perpetuation of the image. Rituals are also equally important to the developer as part of the IS profession, as they provide order and structure to what, without them, may appear random; moreover, they are a key ingredient in the expression of professional solidarity (Trice and Beyer 1984).

As the high priest of technology, the developer possesses the apparent magical quality of making the computer productive, transforming a highly unintelligible piece of technology (to the lay person) into a key organizational tool. The magic is further enhanced by the separation of the high priest from the laity (the users). This separation involves a disparity in both knowledge and authority. Because of this disparity, the laity perceive the priests to possess magical qualities. They also therefore look to the priests for guidance and leadership (cf. Mowshowitz 1976, Boguslaw 1965).

2. *Expert Systems as the Embodiment of the Human Expert.* Expert systems have been portrayed as vehicles for substituting a computer for a human expert; they can reproduce the knowledge possessed by experts or professionals. For example, one of the stated reasons for building medical expert systems is to provide advice to lower level medical staff where the use of full time physicians would be impractical (as might be the case in submarines, Rogers et al. 1979). Expert systems have been shown to work well in domains which are well structured and where the knowledge base is reasonably clear (Barnett 1982, Blois 1980, Hayes-Roth et al. 1983, Jackson 1986, Leonard-Barton and Sviokla 1988). While the limitations of expert systems are generally recognized by computer professionals, to the lay person they give the impression of intelligent behavior. This "intelligence" has a particular magical quality about it because it is possessed by an inanimate object. (Stevenson (1987) provides an excellent case of how expert systems in the UK financial services sector have taken on such a magical quality.) The system to mimic a therapeutic encounter between a patient and a psychotherapist developed by Weizenbaum (1966) called ELIZA was plausible enough to suggest intelligent behavior to several lay subjects. Although the system was not originally developed as an expert system (it predated the advent of expert systems by almost a decade), it was viewed by the subjects as embodying a human expert. The magical quality of a computer program (the expert system) mimicking a human expert such that it is perceived as possessing human characteristics is viewed with disquiet by a number of commentators (Weizenbaum 1976, Dreyfus 1972, Mumford 1987).

Other examples of magic within the context of information systems can be found in Stamper's (1985) notion of "information as mystical fluid," Scarrott's (1985) contention that "information is the life blood of organizations," Shallis' (1984) image of the "computer as idol," Brooks' (1987) contention of "no silver bullet" for software development, and Hoare's (1984) classic concern "programming as sorcery or science."

3. Evidence of Symbolism in Information Systems Development

In order to illustrate symbolism in ISD, we present a series of excerpts from six case studies conducted over the last decade (Newman 1989). Here, no attempt is made to demonstrate all aspects of symbolism in ISD. But through an interpretation of the records from the case study interviews, we see numerous instances of myth, metaphor and magic indicating the richness of using symbolism for understanding ISD.

These cases involved private and public organizations, some of which were commercial and some nonprofit. The sites included banking, insurance, financial services, utilities, and universities and were drawn from the U.S., Canada and the U.K. The organizations were chosen on the basis of the following criteria:

1. The organization was introducing (or had recently introduced) a major information system which crossed traditional organizational boundaries. The scale of the project would thereby ensure that if ISD issues are important they will be highlighted in such projects.
2. Sufficiently high-level access could be obtained to enable the researchers to observe and interview subjects in several locations or departments.

Once access had been negotiated, the researchers attempted to identify key subjects at different levels in the development process, including systems staff, user personnel, user managers, and nonuser decision-makers. This enabled the researchers to obtain

multiple perspectives of the same process. As research progressed, further subjects were suggested by existing personnel. A nondirective interviewing technique was used, as this allowed the respondents to express their own views in their own words rather than force their experience into predetermined categories. However, the interviews focussed on certain topics which are important in the literature (user involvement, top management support, power, politics, etc.). Subjects were also encouraged to illustrate the development process through *critical incidents*, episodes which subjects believed were crucial to the success (or failure) of the system. The interviews were tape-recorded, and verbatim transcripts were produced from the recordings. The evidence here is presented mainly from four of the most representative cases. (A detailed description of the cases and the research methodology is contained in Hirschheim, Klein and Newman 1989.)

3.1 *Evidence of Myth in Information Systems Development*

The six myths noted in §2.1, where we provided evidence from the literature, are now revisited using extracts from the case studies.

1. *User Involvement Is Beneficial and Is to Be Encouraged.* While there is an apparent, strong belief that user involvement can bring benefits, we present some evidence which suggests that its practice may raise as many problems as it solves. Not only can it be subject to manipulation by analysts or designers, making it more of a weapon than an aid, but it can also precipitate conflict among the user groups which must then be carefully dealt with. Both these phenomena were illustrated in the case studies.

a. *Manipulation.* In the course of involving users in design, some analysts cannot resist the temptation to manipulate. Here an analyst at a wholesale company candidly describes his approach to user involvement with one user in particular. Firstly, he acknowledges the user's importance:

Analyst: The best designed system in the world will bomb completely if they don't want it and even with these "little people," and that is not meant derogatively. . . . Oh yes. I had one person, one female, who had been with the company 10 years at that time, and she was the top person. Boy did she resist! Took a long time to win her over. . . . I kept asking her advice. . . . I had already figured it out but I wanted her to do it. . . . You have to work on these people, *butter them up*. . . . You are actually sometimes designing it yourself but let them think they did it.

This extract shows how such a commonly advocated technique like user involvement can easily dissolve into manipulation. Although he believes the users to be a homogeneous group ("little people") he skillfully identifies one of the employees as key to ISD success. He then manipulates the relationship to achieve what is a preconceived design ("butter them up . . . let them think they did it").

b. *Conflict Generated by User Involvement.* One of the unintended consequences of user involvement is the conflict it sometimes generates or precipitates (Robey, Farrow and Franz 1989). Forcing hostile people to cooperate in designing an information system can reveal conflicts which have lain dormant for years. In the case of a large state university which was trying to implement an on-line, university-wide admissions system, the process of user involvement brought sharply into focus some of the differences between departments. These departments had, up to that point, acted in a largely independent fashion and had each developed unique procedures for dealing with admissions. Over two decades, these loosely-coupled departments (cf.

Weick 1976) had developed their own batch processing systems. They were now being requested to cooperate in a corporate, uniform design which would bring many benefits to the university if only a few to the departments. There were many examples of where the new system required that the users across several departments sit down with each other and debate policy issues. This was often a painful experience for the participants because it forced the department administrators to make explicit their policies on admissions. These often-vague procedures, which had co-existed comfortably for many years, were largely irreconcilable. For example, the issue of students' addresses forced the different units to sit down with each other and define a student's address. There turned out to be 13 possible unique addresses a student could use. However, the issue that precipitated the most conflict was who should have the authority to change addresses. In the following extract one of the directors of admissions describes the process vividly:

Director: If you want to see a fight or you want to ask a question that will give you an example of the tenderness, ask "Who can change somebody's address?" I have never seen so much argument over addresses. . . . There have been knock-down graveyard fights over who can change an address.

Clearly, from the users' viewpoint the issue over addresses (and this was only one of several issues raised) had generated considerable friction among the user groups. This is indicated by the terms reported—"fight," "tenderness," "argument," "knock-down," "anger". The new system was acting as a catalyst for this conflict.

User Liaison: It is forcing them to interact with each other in ways they have never thought about before, and that has been fraught with frustrations up to this point, and I suspect that we'll have much more frustration as time goes on. And whether or not it will even wind up being worth it is not clear.

User involvement may bring many benefits but, as can be seen from the examples presented, it can also be accompanied by significant "costs." Much of the conflict generated may be unavoidable, especially if an organization seeks to impose a corporate approach to information processing. Nonetheless, the consequent resistance and entrenchment need to be managed by the project leader before it proliferates.

2. *Resistance to ISD Is Dysfunctional and Should Be Eradicated.* The strength of the myth that resistance to ISD is dysfunctional was apparent in several of the case study records. We have already noted that the response of the analyst at the wholesale company to user resistance was to manipulate the relationship, giving the impression that he was involving the user. Although resistance to ISD may be inevitable in any medium to large-scale project, the methods of dealing with it vary. In certain instances, resistance is active and takes place during development. We term this type "preventive" resistance. In other situations, resistance occurs after implementation of the system. This we call "post facto" resistance. Both are described below.

a. *Preventive Resistance.* As an example of preventive resistance, at the state university the project leader was becoming increasingly frustrated by the admissions staff. The users were being asked to accept a multi-screen design which was to be general enough to be used by all admissions units. This meant that for each student processed, admissions would need to flip between eight to 10 screens. Moreover, each screen change was taking around 60 seconds. This was unacceptable to the users who *knew* operationally that they would be in severe difficulties if the design went ahead:

Technical Coordinator: We rebelled against that. The reaction of the systems people was "Oh, it's going to go like this. It's going to go really quickly once you get used to it, but you are just nervous about it." . . . The screen change was so slow. We would sit for over 60 seconds and wait for the screen to change, and we were hysterical. We were hysterical because all you had to do was . . . multiply this by eight screens by 2000 applications and we would be dead in the water.

When this elementary attempt at consensus building failed (as quoted above by the user), the project leader moved quickly to more coercive techniques:

Project Leader: . . . In Undergraduate Admissions, one of the people there said "We don't really want it. We've talked it over and we don't really want this new system. . . . We resist the change. We don't want to spend the time working at this pace with you. We don't want to give." And basically the computer center says "Well, look, you have no choice." I guess this is not the computer center. I guess maybe the mandate comes from higher level: "The old system will be turned off and Thruway will be here to stay and you're not going to have that choice."

Although the users resist, they are delivered an ultimatum by the computer center ("you have no choice" and "the system will be turned off"). When breakdowns in communications such as this occur between systems and user personnel, it is a small step to use raw power to "clear the path" (cf. Markus 1983, Newman and Rosenberg 1985, Keen 1981).

In this case, the issue over screen design did not turn out as the project leader had hoped. It was clear that neither systems nor admissions could resolve this issue at the ground level. Neither the project leader nor the technical coordinator had the authority to insist on their way. So the problem was pushed up to the admissions director who issued an ultimatum. The users got their single customized screen (although it was not on-line). Once the reasons for confrontation were resolved, the possibility of genuine cooperation emerged. However, the result could have been quite different if the systems staff had coerced the users and forced them to proceed with the original multi-screen design. If the system had gone ahead, the "victory" for systems would have been short-lived. Admissions' staff, with their "private," inside knowledge *knew* the system was not operationally feasible. Although they could have entered each applicant's details into the system, the volume of applications and the time to process each one (30 minutes) would have overwhelmed them. This was obvious to the users but not to the systems staff, who failed to grasp the magnitude of admission's task. Hence the users' resistant stance. If the design had gone ahead as originally proposed (i.e., unworkable), the subsequent problems could have easily brought the whole project to a messy end.

b. *Post Facto Resistance.* In other cases, the resistance may be far more muted and may occur after the system is implemented, and thus not be addressed at all by the technicians. For one medium-sized insurance company, the computerization of personal lines insurance had been largely successful. When it came time for the commercial underwriters to begin using their system, the result was far less successful with many underwriters either ignoring the system or using it minimally for documentation.

Underwriter: A lot of things were getting jammed and that's where some of the unresolved frictions emerged. And now you are not only being controlled by the home office . . . you've got administration controlling you, because they are shipping out the policies for you

without you requesting them. . . . We used to try every chance we could *not* to put it on the computer. We would say, "sorry, this can't go on the computer."

The underwriting system was cumbersome and awkward to operate. Because the underwriters felt unable to change the system they simply stopped using it. Although the resistance in this case was passive, it was nonetheless a sign of problems with the system.

It is our belief, therefore, that *users' resistance is not always a behavior that must be eradicated*, as is often supposed in the management literature, and observed in the university case cited above. Moreover, even where the resistance arises after the system is implemented as in the insurance underwriting case, the developer needs to be aware of passive (post facto) resistance. In these cases, as Markus (1983) also noted, the signs of resistance signal the designer that further investigation is warranted.

In conclusion, the analyst on a project can choose to focus on techniques and technologies to overcome resistance, as if resistance were a behavior to be suppressed or eradicated. On the other hand, the analyst can try to understand the reasons for the resistance and treat such signals as signs of the organizational stresses induced by the system changes. Indeed, the signs of resistance might show that the proposed system would be unworkable if pursued, as in the university example.

3. *Information Systems Should Be Integrated Wherever Possible.* The integrative motive among systems designers forms a strong influence in ISD. Like many of the myths of design, it offers enormous potential in rationalizing organizations, but the delivered product can fall well short of the ideal. In the university, each of the admissions units had developed their own independent systems. These were batch processing systems where admissions data is accumulated over a pre-defined period (day, week, etc.). The processes would vary from one unit to another because each had developed its own system to meet its own needs.

The drawbacks of such systems are well documented (e.g., Everest 1974). Apart from poor timeliness, the information produced by different systems would inevitably be incompatible, making it extremely difficult to compare admissions data between units as would be required in a corporate approach to admissions and recruiting. In contrast, the new system would be on-line and integrated with the rest of the student information system: as admissions data are entered into the database, the files would be instantaneously updated. Access to the database would, therefore, provide the latest information on students and, because data would be stored in a standard format, reports could be readily extracted. The project leader here compares the two approaches:

Project Leader: You want to get an integrated system, so that all of the systems have one database, a biographical database, and have an integrated system for financial aid, billing, Thruway (admissions) and eventually student records. We do not have that. That will never happen on their old way. Right now they're able to process their applicants fine. But if it's in the best interests of the university to have an integrated system, that won't work, not with their old batch systems, isolated systems.

The admissions function at this university was previously a collection of independent data entry and processing systems (undergraduate, graduate, law, etc.). The new design was a corporate approach to admissions, a radical change from the status quo. But in order for the corporate approach to work, the different groups had to cooperate. Users, on the other hand, did not generally have an integrated view of systems

and organizations. They largely viewed systems bottom-up and did not appreciate the corporate advantages continually stressed by the project leader.

Similarly, in the insurance company, the 18 branch offices had been run largely autonomously. The type of business each office wrote was a matter for them to determine. Under the manual system, the commercial underwriters enjoyed a considerable flexibility in work procedures. Documentation, a key task in underwriting, was a good example. When an underwriting decision is made, it is not always possible to document the process because of the pressures of other work:

Underwriter: If you have an account that you know you've not done proper or you haven't had time to fully document though you know your decision is good, they hide those.

Interviewer: So there are ways of hiding the problems.

Underwriter: I wouldn't even call it a lot of times the problems. It was just if you were working on a couple of large accounts at once and your underwriters really didn't have time to do a terrific job actually hand-documenting the decisions say that you and he or you and she came up with, you put that off and do it later, just to keep the paper moving.

When the head office auditors came to verify that procedures were correctly followed, the underwriting staff could offer up accounts, together with any explanations in an accompanying report as a way of "embellishing" them. With the new system, head office had control of the underwriting database. Instead of offering up accounts for auditing purposes, the head office auditors could select the ones they wanted with no chance for the underwriter to offer written or verbal explanations:

Underwriter: Now what was happening was that they were pulling this stuff up on the computer with no explanation.

Interviewer: No way of you interpreting.

Underwriter: And calling down, "What the hell is going on here? I see your increase. You just had a production of 20%. You know we've kind of asked you to stay at 15." Whereas having been able to write a report, you could say, "Well, gee. One account just went on the books from the last quarter," and such and such. . . . And it always seemed to work well. But now we were being put in a defensive position instead of offensive in that way. The other thing they used to do a lot was they'd come down and audit. . . . The manager used to send them a list of our larger accounts or whatever mix they wanted. Now they just pulled it off the computer.

It was clear that the new system, designed as a centralized strategy to underwriting, did not "fit" the organization's decentralized structure. Using a head office database afforded a degree of integration of procedures unknown before in the company. However, the system, as we have already seen, was rejected by some of the underwriters who refused to use it or used it only minimally.

In summary, organizations may desire the advantages that greater integration apparently offers. However, this strategy needs to be weighed against the dysfunctional consequences of pursuing greater rationalization and integration. The cases cited above indicate that the solution can sometimes become the problem.

4. *The Systems Developer Is Generally the Best Person for Making Decisions About the System.* The myth of "we know best" is particularly well illustrated in the gulf in attitude between systems personnel and users. Users at the university had misgivings about working with the systems people. In the following extract from an interview, the technical coordinator from admissions shows the depth of her feelings:

Interviewer: How did you react to people who were trying to get the specifications from you? What kind of relationship did you enjoy with the systems people?

Technical Coordinator: We were very defensive, because we felt they were trying to have us fit the system and we wanted to have the system fit us. . . . We were convinced they weren't after our best interests and they were convinced we weren't after theirs. So that we were just against anything new. We had quite a time of it.

We felt that they didn't have our best interests in mind, they had their best interests in mind. We literally didn't believe a word they said.

The analyst at the telephone company was an exceptional person by most standards and enjoyed excellent relationships with users. Nonetheless, he summarily dismissed any suggestion that human behavior was complex: he held a very simple model of human behavior:

Analyst: I have very little time for all this psychological stuff, as I call it. . . . [A person] is a leader or a follower. To me there are two categories that make sense. . . . That's the way everything has worked for two million years.

The project leader on the Materials Management System at the telephone company was even more dismissive than the systems analyst concerning the importance of users, some of whom were left uninformed about impending changes:

Project Leader: . . . To me this . . . [is] just another tool. Using the tube is very simple. And this thing is just another tool.

Interviewer: Are they [the clerical staff] at the moment fully aware of what's going to happen and how the change is going to affect them?

Project Leader: No they're not.

The attitude of systems staff has been consistently cited as one of the causes of bad relations with users whose frequent response is to resist the system's development. In the case of the university, the differences were clearly seen in the interests of each group. The systems group was perceived to be only concerned with implementing the system as originally designed. In contrast, the users' interest was to get their task done. The result was poor communication and distrust ("We literally didn't believe a word they said").

5. *Politics Should Not Be the Concern of the Systems Developer.* The myth that "politics should not be the concern of the developer" had wide credence in our case examples. At the university, the central arguments of the users were about resources and what effects the proposed system was going to have on their domain. The project leader constantly stressed the corporate benefits of the new admissions system:

Project Leader: Right now they're able to process their applicants fine. But if it's in the best interests of the university to have an integrated system, that won't work, not with their old batch systems, isolated systems.

She seemed perplexed by the reactionary stance of the users, clinging to their "old batch systems, isolated systems" like ancients clinging to their idols. An appeal to university goals seemed to fall on deaf ears. But from the users' perspective, this was entirely reasonable. Indeed the analyst herself acknowledged the effectiveness of their existing systems ("They're able to process their applicants fine"). In contrast, the director of undergraduate admissions demonstrated her political acumen in dealing with the systems group. She realized that the new system would require her staff to perform much more data entry work. In negotiating with the systems people, she was able to campaign effectively for additional staff:

Director: I said to them, "we'll try, but if we can't do this, we're going to batch it over." And they say, "Oh no, you can't. What's the point of an on-line system if you're going to batch the data?" This was after the Director had said to me that he didn't see why I would need additional staff people; that existing staff wouldn't have to code things now, they could just input directly. So I said, "That's fine. We'll try that and we'll do our best, but if we can't handle the volume, then we'll be batching over." So the next thing was that we'd be needing additional people to help.

Here the director of admissions, faced with a system which she considered to be unworkable, does not reject it out of hand ("We'll try that and we'll do our best"). Instead she offers what she knows to be an unacceptable solution to systems staff ("We're going to batch it over"). As we have noted in previous comments, this would have meant abandoning the on-line philosophy and a return to 1960's batch technology. The director's final comment is spoken as if she is reporting a remark by the systems staff. The additional staff which admissions would need for data entry would now be forthcoming.

For the insurance company, the politics were concerned not so much with conflict resolution but with conflict avoidance. The new system was an accepted fact, but nonetheless, the job of underwriting needed to have priority. The branch realized they were not strong enough to fight the systems group:

Underwriter: And you became good at circumventing [the system]. Because it becomes a political ball game. You can't undermine what the computer department is trying to do, yet you know that you are in for short-term, long-term strategies—you've got to survive. And you can't really be subversive to their efforts, because it comes back to haunt you anyway, but you have to do what you have to do to get your work out. You can't complain. It's one of those things you do quietly. Everybody found different ways. Maybe they won't find out. Maybe they won't ever see it.

Even their manager was a party to circumventing the power of the computer department:

Underwriter: You know, it was one of those situations where you really could not bring it up in your department meeting that you were having these problems. . . . I really think [her manager] must have known that we were not using it to its capabilities. I mean he had to have known. Politically, he could not come out in the meeting and say, "Yeah, I support you." I think he more or less let it go, and then if a problem arose he would deal with it. Again, all crisis management.

In the illustrations above, we see both the ineffectiveness of avoiding politics on the part of the systems developer and the willingness of the users to participate in the political process which they saw as a natural part of organizational life.

6. *The Key to Successful Design Is the Use of a Top-Down Approach.* To some systems staff, the top-down approach to systems development offers a mythical quality to the design of systems. In the telephone company, the project leader outlines the approach used for the materials management system, noting the normative nature of the top-down approach.

Project Leader: We're following a certain approach, a top-down approach, and what we've done is define all the logical functions of the system first so we haven't cluttered our minds up with how it's currently being done, or who is doing it, or where it necessarily is happening. . . . That is, the approach we're taking, is the totally top-down approach. That's how you have to design and work on management information systems.

Note how the ideal approach leads him to ignore the current organizational realities which "cluttered our minds up." But he notes the approach is not without its problems for the users.

Project Leader: Top-down is a troublesome approach for the users because the user is used to his function and so that means his view of looking at it is bottom-up really. Now when we look at it top-down, especially in the initial stages, he's busy saying "that's a function but where does my thing fit into this?" He's trying to fit his bottom-up view into this top-down and it doesn't fit very well because he isn't going to get his bottom-up [view] until we get through the methods and procedures. While we're at the stage of designing the logical system, it's a very troublesome thing for him to relate to.

Nonetheless, the project leader returns to the "advantages" of the approach for those who will use it even if reality (e.g., conflict of interests) occasionally creeps in:

Project Leader: I think their concern is the current people that are here and if we're going to give a system that is going to give job enrichment to those people, its going to free them of the number crunching that they currently do, it will give them more tools to be effective in their jobs. And it's going to increase their productivity in a painless way. Why the hell should they be concerned, they should be happy, they should want to be part of this thing and endorse it. Naturally, things don't work out like that in reality. You get a conflict of interests, perverse things creeping in, little rules and regulations, glitches, and all the rest.

Like many projects which employed this approach, the project was never completed and cost the organization over \$2 million. While it is rare to observe this myth in its pure form, it appears to be widely held among developers. And as this case shows, it will often manifest itself in a series of organizational contradictions.

3.2 Evidence of Metaphor in Information Systems Development

In the case studies, a number of powerful metaphors were used by various ISD participants. The following are three such metaphors, only one of which has not been discussed in our literature review of §2.2.

1. *Information Systems Development as a Battle.* As previously discussed in §2.2, military metaphors are particularly common not only in everyday experience but also in systems development. For example, a senior systems analyst at the telephone company likened systems development to a battle between the analysts and the users. He found that in dealing with some resistant managers, the only way to handle them was to get rid of them. If this wasn't possible, he found it necessary to bring in the "heavy artillery" of a higher authority:

Systems Analyst: A prime requirement of this type of project is to have a very high level, naturally-respected [manager]. . . . You have got to have a lot of weight to fling about unfortunately, in a lot of these cases, to try things out. You can't just force them on the users. Obviously it has got to be done with agreement, but at the same time, where you run into these problems, you have got to be able to rely on somebody to *clear the path*.

Note how the concept "clear the path" can be interpreted in its military connotation just as a general may call on fire power to flush out the enemy. The context of this example was a succession of conflicts between the systems analyst and various user managers which resulted in top management removing several resistant managers from their posts. The analyst clearly saw the organization as a battlefield. Similarly, in the university context we observed the clear drawing up of battle lines. The systems coordinator demonstrates this in the following extract:

Systems Coordinator: So questions on screen design were resolved by the users getting their way. . . . There was an issue on how many data entry screens were too many. And they *won the battle*. We simply redesigned data entry screens.

Although we have only shown two instances of this metaphor, its occurrence was widespread in our data indicating that "battle" needs to be given serious

consideration in further studies in this area. This is a point also emphasized by Lakoff and Johnson (1980).

2. *Organizations as Fiefdoms.* It is possible to conceive of organizations in terms of many different metaphors (see for example Morgan's (1986) eight organizational metaphors). For example, in the university case, one particular user manager interpreted his own organization in feudalistic terms:

User Manager: In terms of fiefdoms. . . . there are different barons responsible for the different parts of the university. And they tend to be very much like many feudal systems. They tend to be very protective of their domain.

In this example, the user manager saw the university as a collection of fiefdoms. He resorted to this metaphor in order to understand the departmentalism and consequent friction which he observed. This territorialism is particularly important for systems which cross traditional organizational boundaries (Newman and Rosenberg 1985).

3. *Man as Machine.* This metaphor is a common one, used in many contexts ranging from the "engineering" of organizations to treating individuals simply as units of production. Here, the project leader at the telephone company assumes that the clerical staff will simply adapt their work patterns to accommodate the new information processing system. The workers are seen as passive components of the designer's new system, and hence, there is no contradiction in keeping them uninformed:

Interviewer: What about at the shop floor level? I noticed there are manual files kept up there; rotary type of files, will all that disappear?

Project Leader: Yes.

Interviewer: And how are you going to handle that kind of changeover?

Project Leader: The rotary file you speak of is a work order file of materials for plan requirements. That will all be contained in the computer. . . . That rotary will be replaced by a different kind of rotary, a dual density disk with all the data stored on it, and then instead of accessing the paper, they'll have a "tube." . . . To me this was just another tool. Using the "tube" is very simple. And this thing is just another tool. The information systems give them the same capabilities of a pencil. The computer enables them to add, delete, or change information within it.

Interviewer: Are they at the moment fully aware of what's going to happen and how the change is going to affect them?

Project Leader: No, they're not.

In this example, workers are treated as largely irrelevant, and hence, there is no need to involve them—only use them. The "man as machine" metaphor is particularly powerful here because it allows the project leader to simplify a messy world by treating users homogeneously, i.e., as machines (Burns 1981).

3.3 Evidence of Magic in Information Systems Development

In the case studies, we often observed the rituals inherent in magic rather than magic itself. In each case the rituals had form but no substance—they were façades. The participants however, were believers—they saw no contradiction between form and substance. The following are three representative examples of magic within ISD, which complement the two examples commonly discussed in the literature and reviewed in §2.3.

1. *User Involvement as Magic.* We have already seen some of the mythical dimensions of user involvement in §2.1 and 3.1. In the following extract, we see evidence of associated rituals. In this example, the analyst from the wholesale company goes through an elaborate ritual in order to convince a key user that her views are significant for the design of the system:

Analyst: I'd say, "So-and-so, I can't quite figure this out. Now, you have more experience than anyone else, what would you do? We want to be able to put into the computer such-and-such piece of information and we are going to use it to produce other reports, and so on, but what is the best way of getting it to the computer?" And let her work out the problem. I had already figured it out but I wanted her to do it.

The ritual hides the fact that the analyst has already designed the system. This type of user involvement has form but no substance; the user has no genuine opportunity to influence the design of the system. In this sense, user involvement is but a façade.

2. *Sign-off as Magic.* Sign-offs are typically discussed as a procedure for letting the user know in advance what the final system design will be. Not only does it allow the systems developer to maintain control of development, but also control of user expectations. His belief is that the sign-off is the manifestation of the users' understanding of the system. The user signs-off only when he understands and approves the system on paper. In the following extracts, however, we see the user neither understands the system nor the implications of her signing-off (a situation which we suspect is not unusual). In the university case, the analyst quoted had spent a long time with the users obtaining requirements (the scope document):

Analyst: The user has been involved right from the scope document . . . and they agreed. So they basically read it and approved it. . . . It took months and months to get approval.

The user saw the process very differently, not realizing the implications for signing-off the system:

User: They introduced us to the system and we created things like scope documents, what we wanted out of the system, what we felt we needed, what it should do for us, staffing issues. Really we were operating blindly because we had very little to no experience with it.

This ritualistic exercise resulted in a compound failure. It did not live up to its purported objective of securing the users' understanding of the system-to-be. But more importantly, it was in fact counter-productive: it helped create, nurture and seal a deep misunderstanding which eventually led to the users' rejection of the implemented system.

3. *Ownership of Data as Magic.* The ritual surrounding the ownership of data is a strong one and can be vividly seen in the university case where one of the user groups was reluctant to pool student data despite having no experience with data sharing on a computer in the past. Here one of the user managers bemoans their attitude:

User Manager: But what I see is that people are very proprietary about the information that they have. "It's not the university's, this is our information". . . . I asked "If we ever had the capability, could we access that data in an on-line environment?" and immediately alarm bells went off . . . the law school got extremely nervous about the idea that somebody else possibly might be able to have access to any type of information on law students.

As this example suggests, it can often be noted that there is a mystical value attached to the ownership of data. The sharing of data is thus to be avoided. There is a

TABLE 1
Case Study/Symbol Mapping

CASE STUDY	MYTHS						METAPHORS			MAGIC			TOTAL
	USER INVOLVEMENT	RESISTANCE TO SYSTEMS DEVELOPMENT	INTEGRATED INFORMATION SYSTEMS	SYSTEMS DEVELOPER-BEST PERSON	NON CONCERN FOR POLITICS	TOP-DOWN APPROACH	ISD AS A BATTLE	ORGANIZATIONS AS FIEFDOMS	MAN AS MACHINE	USER INVOLVEMENT	SIGN-OFF	OWNERSHIP OF DATA	
UNIVERSITY	X	X	X	X	X		X	X			X	X	9
INSURANCE		X	X		X								3
TELEPHONE				X		X	X		X				4
WHOLESALE	X									X			2

strongly-held belief that harm will come from others accessing “our” data, often without any basis in experience.

Table 1 summarizes where the empirical evidence for the various myths, metaphors and magic came from, i.e., from which case studies. It can be seen that each of the 12 symbols presented in this paper were represented in at least one case study, and six of the instances were represented in two case studies. It should be noted that the university case was the source of nine of the symbols. This apparent imbalance is due to the in-depth nature of that particular case study material. Furthermore, because of the loosely-coupled nature of the organizational departments at the university, conflicts were more overt.

4. Discussion

It might be argued that many IS professionals realize that information systems development is not the highly rationalistic exercise portrayed in IS texts and trade publications, and that successful developers have always been conscious of the importance social interaction plays in systems development. However, the large number of IS failures acts as a constant reminder that such professionals are likely to be in the minority. While many would agree that a considerable part of ISD is based on myth, metaphor and magic, there is less agreement on what to do about it. Symbolism may play an important role in systems development, but just noting it does not necessarily lead to better or more successful systems. There is a need to translate such knowledge into practical application—to suggest how traditional systems development might be modified to take symbolism into account.

In the case of *myths*, we can see some dangers if system developers cling blindly to these without considering the particular situation they face. For example, if resistance to information systems is always seen as a negative feature of development which must be contained or eliminated, the designer will never then see the opportunities

for “reading” the signals from the users. In the examples cited above, the users were not blindly resisting the systems; they were, in their different ways, showing that the designed product was flawed in some significant way. In the university case, if the developer had taken these signals of resistance seriously, she would have avoided a great deal of unnecessary conflict and stress, and at the same time, produced a better system, one which would have been acceptable to the users. As another example, politics is often seen by developers as an organizational problem, and thus not the concern of systems staff. In contrast, we have shown that for large projects the systems developer needs to be able to enter meaningfully into the negotiations required to implement systems. These negotiations will be both between systems personnel and users and between the different user groups. Far from disdaining politics and political behavior, such circumstances require developers to become politicians themselves.

While there are many *metaphors* used in ISD by both users and developers, the one that surfaces frequently is the military one of “battle” (cf. Lackoff and Johnson 1980). This is significant because it indicates how often development of information systems is conceived of in confrontational terms. Thus expressions such as “battle,” “winners,” “losers,” “clearing the path” are found, and appeals to formal authority structures are often resorted to. It appears that developers, and users to a lesser extent, often enter the design process with the expectation that conflict will occur, and thereby enact such behavior. While some conflict can be constructive and result in improved design, the win/lose situations typically benefit only a small number of groups of “winners” in organizations. As in the case of the insurance underwriters, the “losers,” if from the user community, will be reluctant users of the system and may never fully cooperate as they retire to “tend their wounds.” Such destructive conflict has great financial consequences for the organization in both time and cost overruns and in unproductive systems (cf. Markus 1983). Moreover, in the long term, reinforcing patterns of “us vs. them” conflict leads to behavior which is difficult to change.

We also observe many *magic rituals* in ISD. These are rituals which have an outward form but no substance: they are *magic*. Although the analyst at the wholesale company believed he was practicing “user involvement,” he went through a ritual which involved tricking the user into believing she was participating in the design. This may be an effective way of proceeding with the design, but it involves distorting communications. From the users’ perspective, she believed that she was contributing from her skills and experience to the new design. This, however, was a façade. She was given no way of influencing the design outcome, as that was preconceived. This appears to be one characteristic of such rituals: they involve distorted communication between analyst and user. If this distortion is allowed to proliferate, we would expect designs which do not match the user’s requirements. Moreover, if the façade is ever destroyed, the users will lose confidence in the developers, resulting in an atmosphere of distrust.

While the above discussion and examples in §2 and 3 illustrate some of the “constraining” influences of symbolism in ISD (e.g., myopia, reification, prejudice, territorialism), there is also another side. Symbolism can be seen in its “enabling” form (e.g., solidarity building, sharing of meanings, value-reinforcing, continuity preserving). For example, previously (§3.3) we showed sign-off as a constraining symbol—a ritual having form but no substance. In contrast, it could be seen as an enabling

symbol of consensual agreement, building solidarity between systems designers and users.³

In order to develop these ideas more fully, consider the issue of user resistance both non-symbolically and symbolically. User resistance has traditionally been interpreted non-symbolically (cf. Hirschheim and Newman 1988). However, an interpretation of how actors use symbols can, we claim, provide new insights about their constraining as well as their enabling influences.

Non-Symbolic Interpretation

When viewed non-symbolically, user resistance is perceived as the normal reaction to change. This is a consequence of a person's inherent fear of change, innate conservatism, inertia and general uncertainty (Ginzberg and Reilley 1957). Typically, strategies to overcome resistance are designed to mitigate the negative effects of change for the users. For example, the use of a change agent is recommended together with "planned change models" (such as the Lewin-Schein or Kolb-Frohman models). In such scenarios, the object is to "unfreeze" current practices. This approach recognizes the reality of resistance which is seen as inertia in the user community. However, users must be educated and structures altered before changes can be introduced, and a variety of techniques have been recommended in the literature. After the change, the new practices can be reinforced as they become established ("refreeze").

Symbolic Interpretation

When viewed symbolically, user resistance is seen as an integral part of a metaphorical "battle"—the conflict between the two opposing sides, the designers and the users. Such an interpretation frequently reveals the constraining influence of symbols. However, we also want to suggest that the same symbol can be seen in its more enabling form.⁴

ISD as "Battle"—A Constraining Metaphor

We choose the metaphor of ISD as battle because of its pervasiveness (§3.2) and the powerful images that it conveys (as vividly portrayed in Lakoff and Johnson 1980). It also relates to many of the other symbols referred to in §3. The metaphor of "battle" as used by systems designers acts as a powerful constraint on behavior. As with all symbols, it shapes the perspective of the designers as they react to their mental images of relationships with users. Thus, the practitioners "enact" their mental models producing all manner of constraining and prejudicial influences and behaviors (Weick

³ Analogously, certain symbols used by researchers can be *constraining*, while others may be more *enabling*. For example, the journal review process is undoubtedly a strong guiding and motivating factor in patterning research activities. One researcher might liken it to a *game* where the "opponents" are journal editors, reviewers, and the promotions and tenure committee members. Another researcher might use the guiding metaphor of *constructive criticism* which might lead to greater theory development and knowledge growth. Although it is not clear that one approach is necessarily more "effective" than the other, it is clear that the different symbols can lead to different behaviors (e.g., those associated with short-term vs. long-term goals). While enabling symbols appear on the surface to be more "positive" and enriching, one needs to be cautious about unilaterally advocating their use. In certain environments and cultures, constraining metaphors may well be more appropriate. Nevertheless, as was suggested in the ritual of sign-off (§3.3), this can be either empty or reinforcing, depending on the actual circumstances and the particular actors involved.

⁴ There are other examples we could have chosen to illustrate both the constraining and enabling aspects of symbols. For example, we could have focused on user involvement as manipulation (a constraining symbol) or user involvement as learning (an enabling symbol).

1979, Morgan 1986). This perspective helps us to understand that systems designers *expect* users to resist systems development. There will also be “winners and losers” in the ensuing conflict. The focus has shifted from organizational development to strategies and tactics designed so that one party can prevail over the other. Therefore, we observe game-playing, manipulation, threats, coercion, and finally, pleas to authority when all else fails (Bardach 1977, Keen 1981). In this scenario, users are seen as territorial-defending their turf, protecting their fiefdoms, etc., and many of these issues have been illustrated in §3 in some detail.

ISD as “Constructive Conflict”—An Enabling Metaphor

If we still accept that there are deeply-held differences in attitudes and beliefs between designers and users, i.e., that there is conflict between the opposing sides, we want to suggest a modification to the traditional military metaphor of ISD as “battle.” Instead of the destructive conflict that researchers frequently report, where stakeholder groups struggle until one prevails, we suggest a metaphor shift to *constructive conflict*.⁵ Using this metaphor sheds new light on user resistance. While constructive conflict may involve some simplification of the ISD process, it does transform many of the issues noted above. For example, when designers meet users there may be conflict (Robey, Farrow and Franz 1989) but the aim could be to reach a consensus on the design, and efforts toward this end could be promoted. In such circumstances, participation would be genuine and not manipulative, and designers and users would be open to learn from each other. This is in contrast to *destructive conflict* where such genuine participation is precluded. Moreover, instead of reacting to the conflicts which occur, designers could choose to create encounters which both reveal and resolve conflicts, for example, through the use of hermeneutic role-playing (Klein and Hirschheim 1983). In contrast to designers dominating ISD, we would see a greater role for user-led designs.⁶ Instead of resolving conflicts by threats, coercion, and pleas to authority, a designer might seek ways of building a consensus with users. In the place of territorialism we would not see a submergence of ideologies, but a mutual respect for differing traditions and customs.

The issue of constraining vs. enabling influences of symbols derives from the inevitability of symbolism in human interaction. When one adopts a particular symbolic view of the world, one inevitably commits to seeing somethings and not to seeing others. Thus, whatever symbols systems developers commit to will enable certain actions as they will inevitably constrain others. In the cases cited previously, certain symbols emerged from the evidence, while others did not (cf. Glaser and Strauss 1967).⁷ In the case of the “battle” metaphor, one that we found in several instances,

⁵ While our use of the term “constructive conflict” may seem somewhat unusual as a metaphor, we believe it encompasses “positive” activities such as team building, cooperation, synergy, and symbiosis which are integral to the notion of such an enabling metaphor for ISD.

⁶ It might be argued that user-led design can arise from a non-symbolic interpretation, as well, because it is a natural consequence of the evolution of the IS function. That is, as the IS function grows, and more and more users expect IS products, users turn to developing (or buying) systems themselves as the IS function can no longer handle the increased demand. We contend such an interpretation highlights the value of symbolism in that it provides an alternative understanding/explanation of systems development phenomena. And it is through such alternative explanations that new insight is gained.

⁷ We acknowledge that we too as researchers are subject to a “paradox of interpretation,” i.e., choosing to see some features of the data as meaningful, while discounting others. As part of a community of scholars, we submit to a process of peer review which itself has enabling and constraining aspects.

we do not know how it came to be held by the subjects nor how it would be discarded by them in favor of enabling metaphors.

While we did not specifically find empirical support for enabling metaphors such as “constructive conflict” in our case studies, some of these enabling symbols are supported by new systems development methods which seek to manifest and articulate differences in the beliefs and attitudes of systems designers and users. The methods attempt in their differing ways to provide an appropriate vehicle for eliciting and resolving the multiple and often conflicting perceptions of the problem domain addressed by ISD. There is a need not only to have more and better user participation within ISD, but to have meaningful dialogue about the nature and type of organizational change which systems development brings about. Development methods such as ETHICS (Mumford 1983), PORGI (Kolf and Oppelland 1979) and other Socio-Technical approaches (Pava 1983, Bostrom and Heinen 1977) move some way toward this, but they are generally weak on specific tools and techniques for attitude and belief elicitation. (Nevertheless, as reported in Hirschheim (1985), participative systems development methods such as ETHICS have been favorably received by the user community.)

Other approaches such as Strategic Assumption Surfacing (Mason and Mitroff 1981) and Soft Systems Methodology (Checkland 1981) offer specific guidelines and tools for facilitating such elicitation. The Functional Analysis of Office Requirements (FAOR) project (Schafer et al. 1988) provides an example of one systems development approach—based on Checkland’s Soft Systems Methodology—which focuses on the multi-faceted and multi-perspective nature of requirements. Through the use of tools and techniques such as “rich pictures,” “root definitions,” and “conceptual models,” the attitudes and beliefs of the various parties involved with systems development are more likely to be brought out into the open. Once these attitudes and beliefs are out in the open, the prospect of resolution increases, although the possibility of conflict due to differences in attitudes and beliefs may increase as well (Robey, Farrow and Franz 1989).

Another systems development approach which accommodates multiple perspectives is the MARS project (cf. Mathiassen and Bogh-Andersen 1987, Lanzara and Mathiassen 1985). Here the focus of attention is on working practices. The starting point is an explicit recognition of the chronic deficiencies of working practices in any group. Various tools such as diagnostic, ecological, virtual and historical “maps” are used to help record and reflect upon the practices (Lanzara and Mathiassen 1985). Maps provide different interpretations of a situation, which help to identify possible ways of acting in this or related situations.

We see approaches such as ETHICS, FAOR and MARS to be better suited for dealing with the symbolic nature of ISD. While such approaches are not panaceas, they nevertheless provide practical vehicles for dealing with the symbolism inherent in ISD.⁸

⁸ While we do not wish to claim that a symbolic interpretation of ISD is the “right” way to conceive of ISD, and the orthodox or rational approach is wrong, it does nevertheless provide an interesting alternative perspective. Such an interpretation is strictly rooted in social anthropology—an area that has been largely ignored by the IS community. It provides a fresh way to think about the problems facing systems development, and may lead to models of social interaction within ISD which would not be possible with our more orthodox conceptions. By adopting a symbolic perspective, we suggest that a better understanding of the nature of the ISD task is allowed.

5. Conclusions

In this paper, we have tried to present evidence from the literature and case studies to demonstrate that symbolism is important in the development of IS. Whether myth, metaphor or magic, we have shown the power of symbolism in *describing* and *explaining* the behavior of developers and users in the ISD process when faced with design tasks that are complex and full of uncertainty. We have shown that symbolism offers simplification, allowing actors to better cope with their world. By patterning behavior and responses to others' behavior, symbolism reduces a messy, complicated world to a simpler one. It also facilitates cohesion, permitting individuals to become accepted members of a group.

Our analysis does, however, point to a number of *dangers* of symbolism in constraining the systems development process. As pointed out in our examples, the myths, metaphors and magic (rituals) employed by developers can lead to inappropriate responses to specific or unique situations. Because it tends to make the developer insensitive to individuals and organizational-specific situations, symbolism can lead developers to make design choices and interact with users in ways which can precipitate dysfunctional (although avoidable) responses from users. They enact the mental images that they use. If "battle," for example, is a frequently evoked metaphor in organizations, systems designers will tend to "see" and expect to see conflict everywhere. While this may be a helpful insight, we also suggest that symbols may be the subject of transformation in organizations leading to alternative approaches to ISD. Thus, for example, instead of accepting *constraining* metaphors such as "battle," design groups could promote and facilitate more *enabling* symbols; for example, as an instance of counterpart to the "battle" metaphor, we explored "ISD as constructive conflict" as a potentially enabling metaphor. We compared the effects such a shift in symbolism might have upon the ISD process, indicating some of the ways of codifying this into systems development methods. It must be said, however, that symbols have an enduring and robust quality which makes them less prone to overt manipulation. Nonetheless, the prospect of mitigating dysfunctional behavior in ISD may appeal to some management coalitions. Consequently, they may attempt to recast symbols into their more enabling form in the hope of building more effective systems. But if managers wish to encourage transformations in symbolism they must be ready to put in place vehicles, such as the emerging systems development methodologies discussed above, to facilitate this process (cf. Smircich 1983). In addition to more general case studies exploring the role of symbolism, we also need detailed case studies of how these transformations might be worked out in practice.*

Acknowledgments. This paper has benefitted greatly from the constructive comments of Kalle Lyytinen, Erkki Lehtinen, Liam Bannon, John King and three anonymous reviewers. The financial support of the Economic and Social Research Council, U.K. (grant no. F0023 2269) is also gratefully acknowledged.

* John L. King, Associate Editor. This paper has been with the authors 12½ months for 4 revisions.

References

- Alvesson, M., "On the Idea of Organizational Culture," Paper presented at First International Conference on Organizational Symbolism and Corporate Culture, Lund, Sweden, June 1984. Quoted in Turner, B., "Sociological Aspects of Organizational Symbolism," *Organization Studies*, Vol. 7, No. 2, 1986.
- Argyris, C., "Resistance to Rational Management Systems," *Innovation*, 10, 1970, 28–35.
- Astley, W. G., "Subjectivity, Sophistry and Symbolism in Management Science," *Journal of Management Studies*, 21, 3, 1984, 259–272.

- Bardach, E., *The Implementation Game*, MIT Press, Cambridge, 1977.
- Barnett, G., "The Computer and Clinical Judgement," *NEJM*, 308, 8, 1982, 494-5.
- Beath, C. and Ives, B., "Competitive Information Systems in Support of Pricing," *MIS Quart.*, March 1986, 85-93.
- Bjorn-Andersen, N. and Hedberg, B., "Designing Information Systems in Organizational Perspective," in P. Nystrom and W. Starbuck (Eds.), *Prescriptive Models of Organizations*, North-Holland, Amsterdam, 1978.
- Blois, M., "Clinical Judgment and Computers," *NEJM*, 303, 4, 1980, 192-197.
- Boguslaw, R., *The New Utopians*, Prentice-Hall, Englewood Cliffs, 1965.
- Boland, R., "Control, Causality and Information Systems Requirements," *Accounting, Organizations and Society*, 4, 1979, 259-272.
- , "Phenomenology: A Preferred Approach to Research in Information Systems," in E. Mumford, R. Hirschheim, G. Fitzgerald, and T. Wood-Harper (Eds.), *Research Methods in Information Systems*, North-Holland, Amsterdam, 1985, 193-202.
- , "The In-Formation of Information Systems," in R. Boland and R. Hirschheim (Eds.), *Critical Issues in Information Systems Research*, John Wiley & Sons, Chichester, 1987, 363-379.
- and Day, W., "A Phenomenology of System Design," in M. Ginzberg and C. Ross (Eds.), *Proceedings of the Third International Conference on Information Systems*, Ann Arbor, 1982, 31-46.
- and R. Greenberger, "Metaphorical Structuring in Problem Formulation: An Exploratory Study," in H. Thomas and L. Pondy (Eds.), *Managing Ambiguity and Change*, John Wiley & Sons, Chichester, 1986.
- and R. Hoffman, "Humor in a Machine Shop: An Interpretation of Symbolic Actions," in L. Pondy, P. Frost, G. Morgan, and T. Dandridge (Eds.), *Organizational Symbolism*, JAI Press, Greenwich, 1983, 187-198.
- and L. Pondy, "Accounting and Organizations: The Union of Natural and Rational Perspectives," BEPR Faculty working paper no. 811, College of Commerce and Business Administration, University of Illinois, Urbana-Champaign, 1985.
- Bolman, L. and T. Deal, *Modern Approaches to Understanding and Managing Organizations*, Jossey-Bass, San Francisco, 1984.
- Bostrom, R. and S. Heinen, "MIS Problems and Failures: A Socio-Technical Perspective-Parts I and II," *MIS Quart.*, September and December 1977.
- Bowles, M., "Myth, Meaning and Work Organization," *Organizational Studies*, 10, 3, 1989, 405-421.
- Brooks, F., *The Mythical Man-Month*, Addison-Wesley, Reading, 1975.
- , "No Silver Bullet: Essence and Accidents of Software Engineering," *Computer*, April 1987, 10-19.
- Bubenko, J., "Information Systems Methodologies-A Research View," SYSLAB Report No. 40, *The Systems Development and Artificial Intelligence Laboratory*, University of Stockholm, Sweden, 1986.
- Buckingham, R. and F. Land, "Education for ISE: What Does it Mean?," *Computer Bulletin*, 3, Part 2, June 1987, 33-5.
- Burns, A., *The Microchip: Appropriate or Inappropriate Technology*, Ellis Horwood, Chichester, 1981.
- Checkland, P., *Systems Thinking, Systems Practice*, John Wiley & Sons, Chichester, 1981.
- Ciborra, C., "Reframing the Role of Computers in Organizations," in L. Gallegos, R. Welke, and J. Wetherbe (Eds.), *Proceedings of the Sixth International Conference on Information Systems*, Indianapolis, 1985, 57-69.
- Cleverly, G., *Managers and Magic*, Longman Publishers, London, 1971.
- Cohen, P., "Theories of Myth," *Man*, 4, 1969.
- Covaleski, M. and M. Dirsmitz, "Budgeting as a Means for Control and Loose Coupling," *Accounting, Organizations and Society*, 8, 4, 1988, 323-340.
- Dagwell, R. and R. Weber, "System Designers' User Models: A Comparative Study and Methodological Critique," *Comm. ACM*, 26, 11, November 1983, 987-997.
- Dandridge, T., "The Life Stages of a Symbol: When Symbols Work and When They Can't," *Organizational Culture*, Sage Publications, Beverly Hills, California, 1985, 141-152.
- , "Symbols' Function and Use," in L. Pondy, P. Frost, G. Morgan, and T. Dandridge (Eds.), *Organizational Symbolism*, JAI Press, Greenwich, Connecticut, 1983, 69-79.
- Deal, T. and A. Kennedy, *Corporate Cultures: The Rites and Rituals of Corporate Life*, Addison-Wesley, Reading, 1982.
- Dearden, J., "The Myth of Real-Time Management Information," *Harvard Business Review*, May-June 1966.

- DeMarco, T., *Structured Analysis and System Specification*, Yourdon Press, New York, 1978.
- Dreyfus, H., *What Computers Can't Do*, Harper & Row, New York, 1972.
- Earl, M. (Ed.), *Perspectives on Management*, Oxford University Press, Oxford, 1983.
- Ein-Dor, P., "Hardware vs. Software Costs: The Composition of Data Processing Budgets," in J. R. Jeffery (Ed.), *Proceedings of the Joint International Symposium on Information Systems*, Sydney, 1988, 107-125.
- Eoyang, C., "Symbolic Interpretation of Belief Systems," in L. Pondy, P. Frost, G. Morgan, and T. Dandridge (Eds.), *Organizational Symbolism*, JAI Press, Greenwich, Connecticut, 1983, 109-121.
- Everest, G., "Database Management Systems Tutorial," in N. Chervany (Ed.), *Proceedings of the Fifth Annual Midwest AIDS Conference*, 1, Minneapolis, 1974.
- Feldman, M., *Order Without Design: Information Production and Policy Making*, Stanford University Press, 1989.
- _____ and March, J., "Information in Organizations as Signal and Symbol," *Admin. Sci. Q.*, 26, 1981, 171-186.
- Franz, C. and D. Robey, "An Investigation of User-Led System Design: Rational and Political Perspectives," *Comm. ACM*, 27, 12, December 1984, 1202-1209.
- Frost, P., M. Moore, M. Reis-Lewis, C. Lundberg, and S. Martin (Eds.), *Organizational Culture*, Sage, Beverly Hills, 1985.
- Gadamer, H., *Philosophical Hermeneutics*, University of California Press, Berkeley, 1976.
- Gambling, T., "Magic, Accounting and Morale," *Accounting, Organizations and Society*, 2, 1977, 141-153.
- Gane, C. and T. Sarson, *Structured Systems Analysis: Tools and Techniques*, Prentice-Hall, Englewood Cliffs, 1979.
- Ginzberg, E. and E. Reilley, *Effecting Change in Large Organizations*, Columbia University Press, New York, 1957.
- Gladden, G., "Stop the Life-Cycle, I Want to get Off," *Software Engineering Notes*, 7, 2, 1982, 35-39.
- Glaser, E. and A. Strauss, *The Discovery of Grounded Theory: Strategies for Qualitative Research*, Aldine/Atherton, New York, 1967.
- Goldkuhl, G. and K. Lyytinen, "A Language Action View of Information Systems," in M. Ginzberg and C. Ross (Eds.), *Proceedings of the Third International Conference on Information Systems*, Ann Arbor, 1982, 13-30.
- Gowler, D. and K. Legge, "The Meaning of Management and The Management of Meaning: A View from Social Anthropology," in M. Earl (Ed.), *Perspectives on Management*, Oxford University Press, Oxford, 1983.
- _____ and _____, "Careers, Reputations, and Rhetoric," Templeton College Management Research Paper, January 1987.
- Habermas, J., *The Theory of Communicative Action, I*, Beacon Press, Boston, 1984.
- Hayes-Roth, F., D. Waterman, and D. Lenat (Eds.), *Building Expert Systems*, Addison-Wesley, Reading, 1983.
- Hedberg, B., and E. Mumford, "The Design of Computer Systems," in E. Mumford and H. Sackman (Eds.), *Human Choice and Computers*, North-Holland, Amsterdam, 1975, 31-59.
- Hedley, A., "Organizational Objectives and Managerial Controls," in J. Woodward (Ed.), *Industrial Organization-Behavior and Control*, Oxford University Press, London, 1970.
- Hirschheim, R., "User Experiences with and Assessment of Participative Systems Design," *MIS Quart.*, 9, 4, 1985, 295-303.
- _____, "The Effect of A Priori Views on the Social Implications of Computing: The Case of Office Automation," *Computing Surveys*, 18, 2, June 1986, 165-195.
- _____, H. Klein, and M. Newman, "A Social Action Perspective of Information Systems Development," in J. DeGross and C. Kriebel (Eds.), *Proceedings of the Eighth International Conference on Information Systems*, Pittsburgh, 1987, 45-56.
- _____, _____ and _____, "Information Systems Development as Social Action: Theory and Practice," *Information Systems Research Center Working Paper*, College of Business, University of Houston, 1989.
- Hoare, C. A., "Programming: Sorcery or Science," *IEEE Software*, April 1984, 5-16.
- Ingersoll, V. and G. Adams, "Beyond Organizational Boundaries: Explaining the Management Myth," *Administration and Society*, 18, 3, 1986, 360-381.
- Ives, B. and G. Learmonth, "The Information System as a Competitive Weapon," *Comm. ACM*, 27, 12, December 1984, 1193-1201.

- Ives, B. and M. Olson, "User Involvement and MIS Success: A Review of Research," *Management Sci.*, 30, 5, 1984, 586-603.
- Jackson, M., *System Development*, Prentice-Hall, Englewood Cliffs, 1983.
- Jackson, P., *Introduction to Expert Systems*, Addison-Wesley, Reading, 1986.
- Keen, P., "Information Systems and Organizational Change," *Comm. ACM*, 24, 1, January 1981, 24-33.
- _____, G. Bronsema, and S. Zuboff, "Implementing Common Systems: One Organization's Experience," *Systems, Objectives, Solutions*, 2, 1982, 125-142.
- Kets de Vries, M. and D. Miller, *The Neurotic Organization*, Jossey-Bass, San Francisco, 1984.
- Klein, H. and R. Hirschheim, "Issues and Approaches to Appraising Technological Change in the Office: A Consequentialist Perspective," *Office: Technology & People*, 2, 1983, 15-42.
- Kling, R. and S. Iacono, "The Control of Information Systems Development After Implementation," *Comm. ACM*, 27, 12, December 1984, 1218-1226.
- _____, and W. Scacchi, "The Social Web of Computing: Computer Technology as Social Organization," in *Advances in Computing*, 21, 1982, 2-90.
- Kolf, F. and H. Oppelland, "A Design Oriented Approach in Implementation Research: the Project PORGI," in N. Szyperski and E. Groschla (Eds.), *Design and Implementation of Computer-Based Information Systems*, Sijthoff and Noordhoff, Amsterdam, 1979.
- Kraemer, K. and J. King, "Requirements for USAC," *Policy Analysis*, Summer 1979.
- Kumar, K. and R. Welke, "Implementation Failure and System Developer Values," in L. Maggi, J. King, and K. Kraemer (Eds.), *Proceedings of the Fifth ICIS*, Tucson, 1984, 1-17.
- Lakoff, G. and M. Johnson, *Metaphors We Live By*, University of Chicago Press, Chicago, 1980.
- Land, F., "From Software Engineering to Information Systems Engineering," in K. Knight (Ed.), *Participation in Systems Development*, Kogan Page, London, 1989, 9-33.
- _____, and R. Hirschheim, "Participative Systems Design: Rationale, Tools and Techniques," *Journal of Applied Systems Analysis*, 10, 1983.
- Lanzara, G., "The Design Process: Frames, Metaphors and Games," in U. Briefs, C. Ciborra, and L. Schneider (Eds.), *Systems Design For, With and By the Users*, North-Holland, Amsterdam, 1983, 29-40.
- _____, and L. Mathiassen, "Mapping Situations within a Systems Development Project," *Information and Management*, 8, 1, 1985, 3-11.
- Leonard-Barton, D. and J. Sviokla, "Putting Expert Systems to Work," *Harvard Business Review*, March-April, 1988, 91-98.
- Lucas, H., *Why Information Systems Fail*, Columbia University Press, 1975.
- Lyytinen, K., *Information Systems Development as Social Action: Framework and Critical Implications*, PhD Thesis, University of Jyväskylä, Finland, 1986.
- _____, and R. Hirschheim, "Information Systems Failures: A Survey and Classification of the Empirical Literature," *Oxford Surveys in Information Technology*, 4, 1987.
- _____, and E. Lehtinen, "Seven Mortal Sins of Systems Work," in P. Docherty, K. Fuchs-Kittowski, P. Kolm, and L. Mathiassen (Eds.), *System Design for Human Development and Productivity: Participation; and Beyond*, North-Holland, Amsterdam, 1987, 63-79.
- Madsen, K., "Breakthrough by Breakdown: Metaphors and Structured Domains," in H. Klein and K. Kumar (Eds.), *Systems Development for Human Progress*, North-Holland, Amsterdam, 1989, 41-53.
- Markus, M. L., "Power, Politics and MIS Implementation," *Comm. ACM*, 26, 6, June 1983, 430-444.
- _____, *Systems in Organizations: Bugs and Features*, Pitman, Boston, 1984.
- Martin, J., *Information Systems Manifesto*, Prentice-Hall, Englewood Cliffs, 1985.
- Mason, R. and I. Mitroff, *Challenging Strategic Planning Assumptions*, John Wiley & Sons, New York, 1981.
- Mathiassen, L. and P. Bogh-Andersen, "Systems Development and Use: A Science of the Truth or a Theory of Lies," in G. Bjerknes, P. Ehn, and M. Kyng (Eds.), *Computers and Democracy—A Scandinavian Challenge*, Avebury, Aldershot, 1987, 395-417.
- McFarlan, F. W., "Information Technology Changes the Way You Compete," *Harvard Business Review*, May-June 1984, 98-103.
- McMenamin, S. and J. Palmer, *Essential Systems Analysis*, Yourdon Press, New York, 1984.
- Meyer, J. and B. Rowan, "Institutional Organizations: Formal Structure as Myth and Ceremony," *Am. J. Soc.*, 83, 340-363.
- Mitroff, I. I., "Archetypal Social Systems Analysis: On the Deeper Structure of Human Systems," *Acad. Management Rev.*, 8, 1983, 387-397.

- Morgan, G., "Paradigms, Metaphors and Puzzle Solving in Organization Theory," *Admin. Sci. Q.*, 1980, 605-622.
- , *Images of Organization*, Sage Publishers, Beverly Hills, 1986.
- , P. Frost, and L. Pondy, "Organizational Symbolism," in L. Pondy, P. Frost, G. Morgan, and T. Dandridge (Eds.), *Organizational Symbolism*, JAI Press, Greenwich, 1983, 3-35.
- Mowshowitz, A., *Information Processing in Human Affairs*, Addison-Wesley, Reading, 1976.
- Mumford, E., *Job Satisfaction: A Study of Computer Specialists*, Longmans, London, 1972.
- , *Designing Human Systems—the ETHICS Method*, Manchester Business School Press, Manchester, England, 1983.
- , "Managerial Expert Systems and Organizational Change: Some Critical Research Issues," in R. Bolland and R. Hirschheim (Eds.), *Critical Issues in Information Systems Research*, John Wiley & Sons, Chichester, 1987, 135-155.
- Newman, M., "Some Fallacies in Information Systems Development," *International Journal of Information Management*, 9, 1989, 127-143.
- and D. Rosenberg, "Systems Analysts and the Politics of Organizational Control," *Omega*, 13, 3, 1985, 393-406.
- Opperman, R., "User Participation: Some Experiences and Recommendations," *Systems, Objectives, Solutions*, 6, 1986.
- Parnas, D. and Clements, P., "A Rational Design Process: How and Why to Fake it," in H. Ehrigg, C. Floyd, M. Nivat, and J. Thatcher (Eds.), *Formal Methods and Software Development (TAPSOFT Proceedings)*, 2, Springer-Verlag, Berlin, 1989.
- Parsons, G., "Information Technology: A New Competitive Weapon," *Sloan Management Review*, Fall 1983, 3-14.
- Pava, C., *Managing New Office Technology: An Organizational Strategy*, Free Press, New York, 1983.
- Pfeffer, J., "Management as Symbolic Action: The Creation and Maintenance of Organizational Paradigms," in L. Cummings and M. Stevens (Eds.), *Research in Organizational Behavior*, JAI Press, Greenwich, 1981.
- Pondy, L., P. Frost, G. Morgan, and T. Dandridge (Eds.), *Organizational Symbolism*, JAI Press, Greenwich, 1983.
- Porter, M. and V. Millar, "How Information Gives You Competitive Advantage," *Harvard Business Review*, July-August 1985, 149-160.
- Raspa, R., "The Refocussing of Folkloric Expression Under Stress in the Automobile Industry," Paper presented at *First International Conference on Organizational Symbolism and Corporate Culture*, Lund, Sweden, June 1984. Quoted in Turner, B., "Sociological Aspects of Organizational Symbolism," *Organization Studies*, Vol. 7, No. 2, 1986.
- Robey, D., D. Farrow, and C. Franz, "Group Process and Conflict in System Development," *Management Sci.*, 35, 10, 1989, 1172-1191.
- and M. L. Markus, "Rituals in Systems Design," *MIS Quart.*, 8, 1, 1984, 5-15.
- Rogers, W., B. Ryack, and G. Moeller, "Computer-Aided Medical Diagnosis: A Literature Review," *International Journal of Biomedical Computing*, 10, 1979, 267-289.
- Rose, M., *Computers, Managers, and Society*, Penguin, Middlesex, 1969.
- Scarrott, G., "Information, the Life Blood of Organization," *Computer Journal*, 28, 3, July 1985, 203-205.
- Schafer, G., R. Hirschheim, M. Harper, R. Hansjee, M. Domke, and N. Bjorn-Andersen, *Functional Analysis of Office Requirements: A Multiperspective Approach*, John Wiley & Sons, Chichester, 1988.
- Shallis, M., *The Silicon Idol*, Oxford University Press, London, 1984.
- Sibley, E., "The Evolution of Approaches to Information Systems Design Methodologies," in T. W. Olle, H. Sol, and A. Verrijn-Stuart (Eds.), *Information Systems Design Methodologies: Improving the Practice*, North-Holland, Amsterdam, 1986, 1-17.
- Smircich, L., "Organizations as Shared Meanings," in L. Pondy, P. Frost, G. Morgan, and T. Dandridge (Eds.), *Organizational Symbolism*, JAI Press, Greenwich, Connecticut, 1983, 55-65.
- Stamper, R., "Towards a Theory of Information-Information: Mystical Fluid or a Subject for Scientific Enquiry," *Computer Journal*, 28, 3, July 1985, 195-199.
- Staw, B., "Spinning on Symbolism: A Brief Note on the Future of Symbolism in Organizational Research," *Journal of Management*, 11, 2, 1985, 117-118.
- Stevenson, H., "Expert Systems in the UK Financial Services Sector: Accounting for the Discrepancies—The Relevance of a Symbolic Approach," Unpublished M.Sc. Thesis, Oxford University, 1987.

- Synnott, W. and W. Gruber, *Information Resource Management: Opportunities and Strategies for the 1980's*, Wiley-Interscience, New York, 1981.
- Tinker, T., "Metaphor or Reification: Are Radical Humanists Really Libertarian Anarchists?," *Journal of Management Studies*, 9, 4, 1986, 363-384.
- Trice, H. and J. Beyer, "Studying Organizational Cultures through Rites and Ceremonials," *Acad. Management Rev.*, 9, 4, 1984, 653-669.
- Turner, B., "Sociological Aspects of Organizational Symbolism," *Organization Studies*, 7, 2, 1986, 101-115.
- Van Maanen, T., "Spinning on Symbolism: Disquisition," *Journal of Management*, 11, 2, 1985, 119-120.
- Vitalari, N. and G. Dickson, "Problem Solving for Effective Systems Analysis: An Exploration," *Comm. ACM*, 26, 11, November 1983, 948-956.
- Weick, K., "Educational Establishments as Loosely-Coupled Systems," *Admin. Sci. Q.*, 21, 1, March 1976, 1-11, 19.
- _____, *The Social Psychology of Organizing*, 2d. Ed., Addison-Wesley, Reading, 1979.
- Weinberg, V., *Structured Analysis*, Prentice-Hall, Englewood Cliffs, 1980.
- Weizenbaum, J., *Computer Power and Human Reason*, Freeman, San Francisco, 1976.
- _____, "ELIZA-A Computer Program for the Study of Natural Language Communications Between Man and Machine," *Comm. ACM*, 9, 1, 1966, 36-45.
- Westerlund, G. and S. Sjostrand, *Organizational Myths*, Harper & Row, London, 1979.
- Wexler, M., "Pragmatism, Interactionism and Dramatism: Interpreting the Symbols in Organizations," in L. Pondy, P. Frost, G. Morgan, and T. Dandridge (Eds.), *Organizational Symbolism*, JAI Press, Greenwich, Connecticut, 1983, 237-253.
- Whitehead, A. N., *Symbolism: Its Meaning and Effect*, Macmillan, New York, 1927.
- Wiseman, C., *Strategy and Computers: Information Systems as Competitive Weapons*, Dow Jones-Irwin, Illinois, 1985.
- Young, S., "Organization as a Total System," *California Management Review*, 10, 3, 1968, 21-32.
- Yourdon, E., *Managing the Systems Life Cycle*, Yourdon Press, New York, 1982.