Emergence in Organizational 'Problem-solving': Theories of Social Cognition

Susan Gasson, College of Computing & Informatics, Drexel University

Please cite this paper as:
Gasson, S. (2006) 'Emergence in Organizational 'Problem-solving': Theories of Social Cognition', Last updated 10/04/2007 12:09:36 ***Working Paper***, retrieved <insert date of retrieval here> . Available from http://cci.drexel.edu/faculty/gasson/papers/probsolv.pdf

Abstract

This paper explores theories underlying the design and development of organizational information systems (IS). It summarizes and compares two very different perspectives on IS design and development: design as the (individual) solution of organizational "problems" and design as the (joint) construction of sociocultural artifacts. The different paradigms of operation engendered by these perspectives is discussed in terms of its implications for information system development (ISD) methods and the management of IS-related organizational change.

Introduction

IS design deals with knowledge-sharing across organizational groups and disciplines. For knowledge to be shared effectively across these organizational "boundaries", we need approaches that view organizational problems as "wicked" problems -- problems that are highly interrelated, hard-to-define and negotiated. Stakeholders from multiple groups and disciplines are required to pool their (often incomplete) knowledge of how the organization works, how they think it should be changed and why, and how information technology should support these changed business processes. This is especially complex when you reflect that people from different parts of an organization (for example, accountants vs. product engineers) often view each other as coming from a different planet! So we need different methods for this type of design initiative -- methods that establish a common language for communication and make explicit the rationale underlying different stakeholder requirements, in a way that stakeholders from other disciplines can understand.

Design As The Solution of Organizational Problems

Design in the ISD literature is often regarded in the context of 'problems':

> " design is to be viewed as the process of problem understanding and problem solving with the aim of producing an artifact." *(Khushalani et al., 1994, page 13)*.

The assumption that problems may be clearly understood and defined in the context of IS design is a misapprehension. Curtis et al. (1988) quote a system engineer, from one of their empirical studies of large development teams:

> " Writing code isn't the problem, understanding the problem is the problem." *(ibid., page 1271).*

This perspective of IS design is also reflected in the organizational management literature:

> " We fail more often because we solve the wrong problem, than because we get the wrong solution to the right problem." *(Ackoff, 1974).*

Lanzara (1983) argues that an understanding of the model which underlies problem-solving behavior in design can explain the core problems which constrain the information system design process in organizations. He identifies three models of information system (IS) design:

1. Design as functional analysis. In this model, design goals and criteria for achieving those goals are pre-defined; the process of design is one of *rationally selecting* means for achieving given ends.

2. Design as problem-solving. The design context contains cues which permit the designer to perform an intelligent search by which the designer *learns* about the structures inherent in the situation and to construct an interface between those structures and the external context. The final solution is just one among many that are feasible.

3. Design as problem-setting. This involves a process of collective inquiry and search taking place through transactions and conversations among several actors with mixed interests concerning the problem at hand. What needs to be created is what the problem-solving model takes for granted: an appropriate problem representation which reflects a decisional structure.

While each of the above models can be found underlying both theoretical and empirical studies of IS design, these three models are insufficient to represent a contemporary understanding of design processes. An additional model is therefore added here:

4. Design as evolutionary learning. The notion that structure is inherent in a situation (as in model 2) is rejected: organizational contexts are seen as dynamic and the objective of design is viewed as an evolutionary convergence between problem-understandings and solution-definitions. This process involves reflective action on the part of the individual: i.e. learning-by-doing, where individuals' courses of action are created and modified by the organizational structures they are acting upon and individuals' actions create and modify organizational structures in turn.

A design context may be characterized by the degree of complexity: the amount of relevant information available in a given situation; and the degree of uncertainty: the availability and reliability of the information available (Matthiassen & Stage, 1992). Each of the models of design is discussed with respect to how it deals with uncertainty and complexity in organizational problem-definition.

Design As Functional Analysis

This model, described as "functional analysis" by Lanzara (1983), is rooted in the scientific management tradition (Taylor, 1947). The rational model of design is based upon a computer information processing model of human cognition (shown in Figure 1), which assumes that all information pertaining to design requirements is available to the designer and that such information can be easily assimilated (Mayer, 1989).

PROBLEM PRESENTATION
--- (apply representational processes)→
PROBLEM REPRESENTATION
--- (apply solution processes)→
PROBLEM SOLUTION

Figure 1. An Information-Processing Model of Problem-Solving (Mayer, 1989)

Psychological models of human problem-solving have been dominated by the metaphor of computer information-processing (Gilhooley, 1989). Representation involves moving from a statement of the problem in the world to an internal encoding of the problem in memory by mentally encoding the given state, goal states, and legal operators for a problem - i.e. by structuring the problem. Solution involves filling in the gap between the given and goal states, by devising and executing a plan for operating on the representation of the problem - i.e. by making a rational choice between alternative courses of action. This model assumes that there is perfect knowledge of design requirements before the problem is structured, as in Alexander's (1964) "synthesis of form".

The functional analysis approach to design involves scientific reductionism (Corbett et al., 1991; Wood-Harper, 1990). A single, technical problem-definition is derived from a 'rational' analysis of organizational goals (Galliers, 1987); this reduces design uncertainty. The removal of organizational and social aspects pertaining to the IS "problem" until the system requirements are defined solely in terms of technical functions reduces problem complexity (Matthiassen & Stage, 1992). While the rational, problem-solving model underlies many of the structured approaches to IS design (e.g. De Marco, 1979; Gane & Sarsen, 1979; Yourdon & Constantine, 1975; Yourdon, 1989, 1993) and has been very influential in forming practitioners' expectations of the process of design, it does not reflect the complexity of problem-solving seen in organizational information system design.

The information-processing perspective uses the 'machine' metaphor to describe the organization: humans may make a rational decision between alternative solutions only if organizational problems are sufficiently structured to be solved by choice between alternative solutions. Mayer (1989) questions four premises of the information-processing model: that humans can pre-determine what course(s) of action are required to reach a given state; that problem representation and solution are independent of each other; that organizational problem-solving can be accomplished by mechanical, algorithmic processes; and that novel problems can be solved by deductive, rather than inductive reasoning. The information-processing perspective is refuted by empirical research, which indicates that designers solve novel problems by generalizing from a similar problem, engaging in random solution attempts or reframing the problem (Lawson, 1990; Mayer, 1989; Malhotra et al., 1980; Turner, 1987).

Design As Problem-Solving

Simon (1960, 1973, 1981; Newell & Simon, 1972) rejected the rational model of problem-solving, with the notion of "bounded rationality". Simon argued that the rational problem-solver is assumed to understand all information relevant to the problem, and to have clear goals and priorities. The concept of bounded rationality accepts that human-beings have cognitive limitations which constrain the amount of information they can absorb and process; the complexity involved in processing and evaluating available information can prevent the individual from selecting the optimal outcome.

Individuals also have access to incomplete information about alternative courses of action, which leads to high levels of uncertainty on the part of the individual. Individuals respond to problem uncertainty by developing a simplified model of the real situation: "bounding" the problem until it becomes sufficiently well-defined to be resolved, they then evaluate alternative solutions sequentially until an alternative is discovered which satisfies an implicit set of criteria for a satisfactory solution. The solution reached is not optimal, but *satisficing*, in that it satisfies a minimal, rather than optimal set of solution criteria. (Simon, 1981).

Simon (1973) describes design problems as "ill-structured" problems. Guindon (1990b) presents a framework for distinguishing between well-structured and ill-structured problems; this is given in Table 1.

Well-structured problems	**Ill-structured problems**
Complete and unambiguous specification of problem	Incomplete and ambiguous specification of the problems
Definite criteria to evaluate the solution and mechanizable process for evaluating if a solution is reached	No stopping rule - no definite criteria to evaluate whether a solution is reached
Any knowledge needed by the problem solver can be represented in one or more "problem spaces"	Many sources of knowledge (problem spaces) that cannot be determined in advance and need to be integrated
Enumerable set of operators that can change the initial state into another state and there is at least one problem space in which can be represented initial state, goal state and all intermediate states	No exhaustive, enumerable list of operators to reach a solution and absence of predetermined solution path from initial state to goal state
Examples: Checkers, Tower of Hanoi, Chess, Theorem-Proving	*Examples:* Design (software, architectural), Planning, Management, Document and music composition

Table 1: Some contrasting features between well-structured and ill-structured problems *(Guindon, 1990b)*

In Simon's model of bounded rationality, individuals decompose an ill-structured problem under the control of a mental, executive process that carries out the necessary coordination functions. Additional information, retrieved from long-term memory, converts the original, ill-structured problem into a collection of well-structured problems: i.e. the process involves inductive reasoning, in addition to the 'rational', deductive reasoning assumed by the rational model of functional analysis. The nature of the problem is no longer unitary - as in the functional analysis model of design - but scientific reductionism is still an integral part of the bounded rationality model of design behavior, as the individual simplifies the problem to reduce uncertainty. The ill-structured organizational problem is viewed as reducible to a set of well-structured sub-problems. Inductive abstraction is required to reduce complexity; Simon (1973, 1981) argues that this behavior is far from the 'rational' decomposition of problem requirements assumed by the functional analysis model of design. In a hermeneutic study of design performed by Boland & Day (1989), a system designer was observed to deal with organizational complexity and political conflict by defining the system in a way which excluded organizational and political issues.

Studies of highly-skilled practitioners indicate that they rely on "intuition" (i.e. inductive reasoning) to problem-solving; such individuals are said to be in the autonomous and most advanced stage of knowing

(Anderson, 1983). Schön (1983) refers to this application of intuitive reasoning as "reflection-in-action". Expert system designers have been observed to apply "data-driven rules" (Guindon, 1990a) - the extrapolation of empirical solutions for similar technical problems - rather than more effective goal-directed behaviors, as data-driven behavior imposes a lower cognitive cost (Anderson, 1983).

Design As Problem-Setting

Wood and Wood-Harper (1993) argue that the use of information technologies has been dominated by the rationalistic tradition discussed in the two categories above; they suggest that design of 'new' technology requires "a focus on the formulation of the 'problem' rather than merely providing an 'objective' description of the problem" (ibid., page 100). But Guindon (1990b) argues that information system design involves the integration of ***multiple*** knowledge domains: the application domain, software system architecture, computer science, software design methods, etc.. Each of these domains represents a problem-space in which a more or less guided search takes place (depending upon which solution paths look most promising and the previous experience of the designer in this domain). The IS development process should encompass the discovery of new knowledge, in particular the discovery of unstated goals and evaluation criteria. Rittell (1972; Rittell and Webber, 1973) defines organizational problem-situations as "wicked" problems. While the concept of wicked problems is similar to Simon's (1960) concept of ill-structured problems, in Simon's (1960) perspective, ill-structured problems may be structured by the application of suitable decompositional analysis techniques - i.e. they may be ***analyzed*** (even if not rationally, in a way that may be justified on rational grounds) - whereas wicked problems cannot be formulated because of their complexity and their interrelatedness with other organizational problems (Rittel & Webber, 1973); they must be ***framed***: defined in terms of contextual frames of reference. A wicked problem (Rittell 1972; Rittell and Webber, 1973) has the following characteristics:

- It is unique.
- It has no definitive formulation or boundary.
- There are no tests of solution correctness, as there are only 'better' or 'worse' (as distinct from right or wrong) solutions.
- There are many, often incompatible potential solutions.
- The problem is interrelated with many other problems: it can be seen as a symptom of another problem and its solution will formulate further problems.

Whereas, in the problem-solving model, problems may be objectively bounded and decomposed, solutions to wicked problems require a more subjective approach: Rittell (1972) advocates 'second-generation design methods' to replace the rational model of design. These methods should include "designing as an argumentative process", which Rittell sees as "a counterplay of raising issues and dealing with them, which in turn raises new issues and so on". This approach is more related to the third model of design described by Lanzara (1983): design as the search for appropriate problem-definitions as well as solutions.

An area of research which explores how designers explore the problem space is the work on 'design rationale'. Echoing Rittel's (1972) advocation of 'second-generation design methods' (i.e. involving argumentation and debate), the ‘design rationale’ theorists (e.g. Buckingham Shum et al., 1996; Lewis et al., 1996; Moran & Carroll, 1996) argue that design requirements are often implicit; they can be surfaced most effectively in the course of social interaction and negotiation if represented explicitly and so made open to inspection and negotiation. The design rationale perspective sees design as taking place within a ‘design space’ - a concept borrowed from cognitive psychology (***c.f.*** Anderson, 1981). The exploration of this design space may be expedited by the explicit representation of design criteria and solution alternatives, using design-space analysis techniques such as the issue-based IBIS technique (Rittel, 1972) or the Question-Option-Criteria (QOC) notation (Maclean et al., 1993).

The design rationale approach sees its objective as recording and understanding the basis of decision-making employed in achieving a (usually technical) design solution. The ***problem*** is defined only inasmuch as the requirements of the designed system are understood and debated by design participants. Whilst the design rationale perspective explores the problem through an exploration of design alternatives and sub-problems and so concentrates upon a problem-space exploration which leads to learning about the solution requirements, the nature of the problem situation and its social and organizational context are ill-defined and remain invisible to the processes of design. Design-rationale goes some way towards the negotiation of problem-definitions: by representing problem-definitions, they are laid open to inspection and negotiation, but these representations are detailed and complex, meaning that non-technical design participants are likely to find their construction problematic, and they are static representations, coping poorly with the emergent and dynamic nature of problem exploration. The approach is seen as most valuable in recording design-space decision-making for the future maintenance of technical systems (e.g. the design of an IT user-interface).

The fifth characteristic of wicked problems (a problem’s interrelatedness with other problems) is characteristic of Ackoff’s (1974) ‘messes’, which are described as "a system of problems" (Ackoff, 1974, page 4). These five characteristics typify the central contradiction of information system design: how may multiple, conflicting problem goals, requirements and constraints be merged into a coherent whole which is acceptable and comprehensible to all those affected by the design - the design team, managers, users, clients and other stakeholders.

Design as problem-setting sees design as a process of collective inquiry and search, taking place through transactions and conversations among several actors with mixed interests concerning the problem at hand (Lanzara, 1983). What needs to be created here is what the problem-solving model takes for granted: an appropriate problem representation which reflects a decisional structure (Lanzara, 1983). While the rational, problem-solving perspective of design attempts to reduce uncertainty through problem-structuring and to reduce complexity through scientific reductionism, the systemic inquiry perspective accepts that complexity is a necessary facet of organizational systems; this perspective manages uncertainty through viewing design as holistic, systemic inquiry into the problem context and manages complexity by recognizing that there are multiple potential target object systems which can be made explicit and subject to negotiation by organizational actors (Checkland, 1981).

The resolution of "wicked" problems in design (Rittel, 1972) is concerned with debate and negotiation to achieve consensus and choice among different, target "object systems" (Welke, 1983); the critical processes are concerned with problem-framing. This perspective argues that there are no objectively given object systems, rather people have viewpoints which enable them to perceive object systems; a multiplicity of viewpoints may prevail among the members of a development team (see Figure 2) and among other stakeholders (Checkland, 1981; Hirschheim, 1985; Hirschheim & Klein, 1992; Lyytinen, 1987). Information system design encompasses multiple perspectives of system objectives, which are ill-defined and open to debate and negotiation (e.g. Mumford, 1983; Checkland & Scholes, 1990; Avison & Wood-Harper, 1990); the subjectivity inherent in this model is encompassed in the use of the term "soft systems" (Checkland, 1981; Checkland & Scholes, 1990).

Lyytinen (1987) proposes that the critical processes of a design team are concerned with achieving consensus on which object system(s) are to be operated upon and the form and scope of the target object system(s).

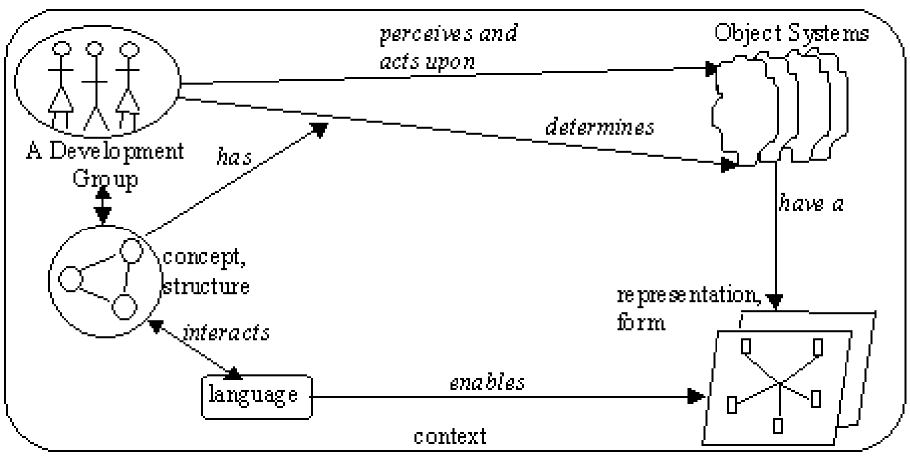

Figure 2. Object Systems In Systems Development (Lyytinen, 1987)

Consensus may mean that perspectives are unitary in nature, reproducing a primary constraint of the previous two approaches to problem-solving. The soft systems perspective is still becoming established: even many writers who have attempted to merge practical approaches to ISD with ‘soft’ approaches speak of a single, ‘primary-task system’ (Wilson, 1984) or the need, early in the IS development process for a "clear statement of system objectives" (Veryard, 1986). Checkland (1980, Checkland & Scholes, 1990) is unclear on this point, but his work has been criticized for privileging the management interest through the search for consensus, which is unrealistic in a political context where management interests dominate (Burrell, 1983). This unitary emphasis can partly be explained by the need for commitments and promises in the negotiated ‘contract’ which represents ISD in practice (Ciborra, 1987) or the need to constrain system requirements in order to meet tight deadlines and resource constraints (Curtis et al., 1988) or because of the individual’s need to reduce problem complexity (Matthiassen & Stage, 1992), but may also reflect the formative influence of rational models of design. Organizational information system design reflects the support of many different tasks and it is the job of the designer - normally a technically trained system analyst - to make sense of the multiplicity of tasks for which support is required and to mediate between the impact of the technical information system being designed and the users of that technology (Boland & Day, 1989). This task involves the resolution of conflicting goals (Methlie, 1980) or multiple constraint satisfaction, defined as "the evolution and testing of part of a design to gradually satisfy its requirements" (Buckingham Shum et al.,1996).

Galliers (1993a) presents an approach based upon SSM, but retaining the plurality of system definitions, while Flood (1995) attempts to deal with the political aspects of systemic design with a "total systems intervention". He suggests that there are three main types of problem-solving methods required for such an intervention: designing, debating and disimprisoning, which are respectively concerned with finding a solution to either efficient processes or effective organizational design, changing people's beliefs and attitudes, and preventing designs and decisions from becoming 'prisons' - i.e. the challenging of received wisdom by questioning whose interests are being served.

For Checkland (1981) soft systems have four main properties: emergence (the exhibition of properties by the whole which are not exhibited by the component parts), hierarchy (entities which can meaningfully be treated as wholes are built up of parts which are themselves wholes, and so on), communication (the transfer of information) and control (the process by which a whole entity retains its identity and performance under changing circumstances). This can be contrasted with the hard systems approach, which sees system properties as being objective, rather than emergent, with communication and control being human interactions with the material (computer-based) 'system', rather than properties of the system itself. While soft systems approaches to IS design see IT as the "serving system" to a "served system" of purposeful human-activity (Winter et al., 1995), hard systems approaches see IT as the target object system. However, this view is still static: the soft systems literature views design as being a process of negotiating a consensus on organizational system definitions, which embody structure and persistence. It may also be argued that the whole thrust of the 'problem' investigation literature in the field of IS is aimed at structuring problems and constructing structured data (Preston, 1991). An alternative model rejects organizational structure as the basis for design (Truex & Klein, 1991): organizations are seem as emergent and dynamic, with design defined as situated, evolutionary learning.

Situated Design: Dealing With Emergent Goals

This section deals with design as the convergence of problem and solution, as distinct from Lanzara's (1983) last perspective of design as problem-setting. Although design is still viewed as being properly rooted in a process of collective inquiry and search, it is recognized that both problem and solution representations do not reflect an appropriate decisional structure, as required by Lanzara (1983), but are emergent and ill-defined; solutions are no longer optimal for that context, but satisficing. There is evidence in some areas of literature (particularly in the field referred to as social psychology) that our conception of design is changing, with an acknowledgement that design is "situated" in organizational contexts (Gasser, 1986; Suchman, 1987; Lave, 1991; Lave & Wenger, 1991). Star (1992) ascribes this change to "the failure of the rational to account for or to prescribe people's behavior" (Star, 1992, page 398). From problem-solving in a rational sense, the situated action perspective views design as a cyclical process of learning about a situation, then planning short-term, *partial* goals (Suchman, 1987), which emerge from the process of design. The nature of the emerging "problem" becomes more complex and unbounded (and, indeed, unboundable) than that assumed in either the problem-solving perspective or the soft systems perspective. Aspects of a solution are explored in conjunction with aspects of a problem understanding; the designer's understanding of both may change as a result of the process. The 'problem' is thus dynamic and constituted of many, interrelated parts.

The evidence from studies of cognitive design strategy indicate that a further model of problem exploration is required to understand design; even the systemic, problem-investigation model discussed below assumes that it is possible to define organizational problems before appropriate solutions can be formed (although the literature is ambiguous on this point, this assumption does appear to underlie the notion of ‘consensus’ in the soft systems literature). But an understanding of organizational problems - and appropriate design goals - *emerges* as partial solutions are explored.

Mayer (1989) demonstrated that representation and solution are interactive processes, i.e. the problem representation is continually reformulated during the process of problem solution. Lave & Wenger (1991) argue that design abstraction is situated in the organizational context: abstract representations of a solution are meaningless unless they can be made specific to the concept at hand. This theory is supported by experimental studies of design (in laboratory conditions), which indicate that designers solve novel problems by generalizing from a similar problem, or by reframing the problem to fit partial solutions which are already available to them from their own or colleagues’ experience (Lawson, 1990; Mayer, 1989; Malhotra et al., 1980; Turner, 1987). Problem and solution are thus interrelated: this concept is synthesized in the diagram of Figure 3.

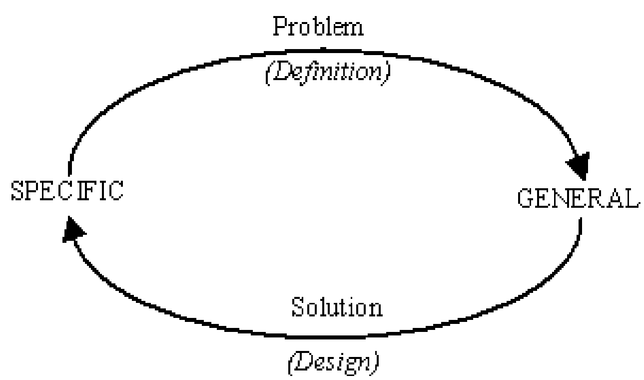

Figure 3. The Inter-related Nature Of Design and Problem Definition

Both Turner (1987) and Malhotra et al. (1980) differentiate IS design from more structured problem-solving by an absence of well-defined goals. In empirical studies of dialogues between designers and clients, Malhotra et al. (1980) observed that designers reframed subproblems to fit an available solution and found that design goals were often partial, implicit and unstated - they were uncovered only when the user stated a system requirement which conflicted with them. They concluded that problem-framing and solution-synthesis were interrelated: problems and solutions converged towards completeness.

So - not only is the problem unclear at the *start* of the process, but the *goals* of the design are also ill-defined in this perspective of problem ‘solving’ - unlike Checkland’s perspective, where one structures the problem situation through defining the goals of a solution. The situated action school of thought considers complication of the problem understanding an aim, rather than complexity-reduction (Boland et al., 1994). Organizations are "organized anarchies" in which people discover analysis and design goals from what they are doing: the processes of bargaining, learning and adaptation (Clegg, 1994).

Turner (1987) argues that "requirements and solutions migrate together towards convergence" and that the process of designing information systems is subjective as well as emergent:

> " Design appears to be more ad hoc and intuitive than the literature would lead us to believe, solutions and problems are interrelated and the generation of solutions is an integral part of problem definition. Problems do not have only one solution; there may be many. Consequently, design completeness and closure cannot be well-defined. There are two categories of design factors: subjective and objective. Objective factors follow from the subjective concepts on which designers model the system. The difficulty in the past is that we have not acknowledged, explicitly, the presence of subjective factors, with the result that, in many cases, objective factors appear to be arbitrary." (Turner, 1987).

Such subjectivity in design problem-investigation is linked with "opportunism" in design (Guindon, 1990a, 1990b; Khushalani et al., 1994). Ball & Ormerod (1995) review the notion of opportunism in system design, which they define as deviation from top-down (decompositional, breadth-first) design approaches and compare opportunistic design with the more structured problem-solving approaches observed in earlier studies of software design. They conclude that much of the structure observed in the early studies of design arose from the more structured nature of the problems set for subjects in experimental situations.

Opportunistic design strategies naturally fit with the prototype or evolutionary systems development approach which permit "learning by doing" (Jeffries et al., 1981). Schön (1983) describes this type of planning through his description of design as "art", which he bases upon the concept of "reflection-in-action"; this concept describes purposeful action which calls on tacit knowledge for its execution. The concept is best described in Schön’s (1983) own words:

> " Even when he [the professional practitioner] makes conscious use of research-based theories and techniques, he is dependent on tacit recognitions, judgements and skillful performances." (Schön, 1983, page 50).

This perspective is contrasted with Simon’s (1973) ‘rational’ problem-solving model by Dorst and Dijkhuis (1996), in Table 2. A critical feature of this perspective is that *each design problem is viewed as unique*: solutions cannot be analyzed, only inductively synthesized from the social constructions of designers.

Item	**‘Simon’**	**‘Schön’**
designer	= information processor in an objective reality	= person constructing his/her reality
design problem	= ill-defined, unstructured	= essentially unique
design process	= a rational search process	= a reflective conversation with the situation
design knowledge	= knowledge of design procedures and ‘scientific’ laws	= the artistry of design: when to apply which procedure/piece of knowledge
example/model	= optimization theory, the natural sciences	= art/the social sciences

Table 2: The rational problem-solving paradigm and the reflection-in-action paradigm contrasted *(Dorst and Dijkhuis, 1996)*

The critical processes of design thus become the exploration, representation, sharing and *evolution* of partial, emergent design goals and the inductive assessment of when a satisficing solution has been reached. This perspective is echoed in the work on modeling design rationale, where the importance of generating and recording subproblems - also referred to as microproblems (Lewis et al., 1996) - is a central concern in making design decision-making visible.

The focus is no longer on the individual designer as *decision-maker*, but on the individual as "*conversation-maker*" (Boland et al., 1994), both through reflective action and through interaction with other stakeholders in the design. Human-beings do not plan actions which are followed through without reflection, but are guided by partial plans which are locally contingent upon the context of activities and material conditions involved in the problem situation (Suchman, 1987). Design problems and partial, ill-defined design goals emerge from the processes of engaging in design activity (Hutchins, 1990, 1991, 1995). This concept can be assimilated with the concept of emergent strategy proposed by Mintzberg and Waters (1985): design problem and goal conceptualizations emerge or are discarded continually through the duration of a design.

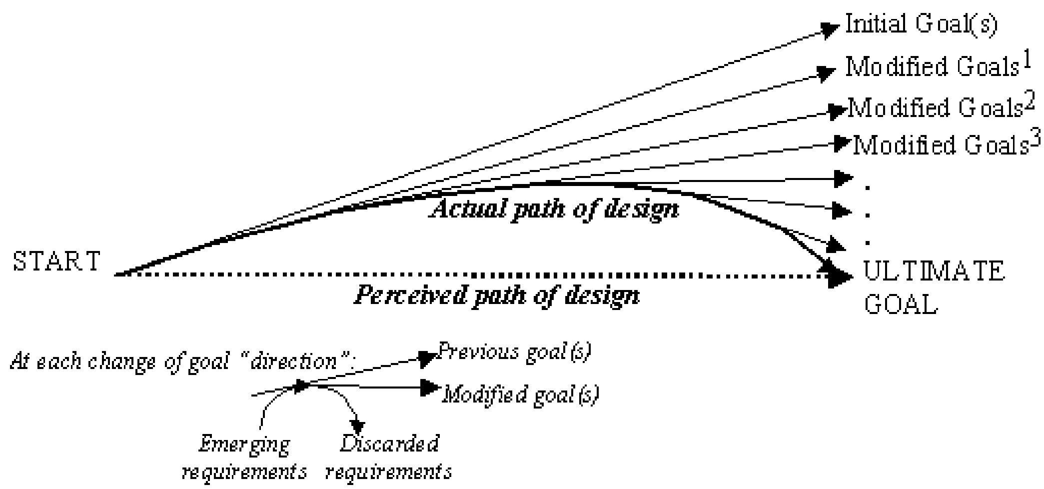

Figure 4. The Implications Of Emergent Design

The implications of emergent design are illustrated in Figure 4. Goals are constantly evolving with an understanding of the design and the actual path of design is much more complex (and longer) than that perceived by actors external to the design process, who only see the start and end points of the design. This model may explain why timescales always 'slip' in IS development projects - a common comment from those not involved in such projects is "why did it take you so long?". A critical process of design must therefore be the management of external perceptions of the design process, particularly those of the "global network" (Law & Callon, 1992) - the informal network of influential decision-makers affected by, and indirectly attached to a design project.

Discussion Of The Critical Processes Of Design

A synthesis of the four organizational 'problem-solving' perspectives discussed above is given in Table 3. It can be seen from this table that the critical issues of design differ, depending upon the perspective of problem-solving which is adopted.

Problem-solving perspective	**Prescribed process for problem resolution**	**Metaphor for organization**	**Assumed nature of problem**	**Nature of problem-**	**Critical processes of design**

				exploration	
Rational (functional analysis)	Decision between all alternative solutions (parallel analysis)	Machine	Unitary, well-structured	Problem solving	Assessing alternative solutions for optimality
Bounded rationality (problem-solving)	Sequential analysis of solution alternatives	Brain	Ill-structured: reducible to well-structured sub-problems	Problem structuring	Assess alternative solutions against minimal, satisficing set of criteria
Systemic (problem-setting)	Exploration of problem situation; achieving consensus on desirable & feasible action	Web	Multiple, inter-related & socially constructed & static	Collective inquiry and problem-search	Shared investigation of and learning about a problem situation; Structuring problems to explore solutions
Emergent (situated design)	Inductive convergence of emergent problem & solution definitions	Organism	Multiple, inter-related, socially constructed & emergent/dynamic	Reflective action: learning-by-doing	Discovering partial, dynamic goals and solutions through experimentation; managing influential stakeholder perceptions

Table 3: Four Perspectives On The Resolution Of Organizational Problems

According to Matthiassen & Stage (1992) the basic characteristics of a design situation may be described in terms of their degree of uncertainty and their degree of complexity, while approaches to design may be characterized as analytical (expressed through requirements specification) or experimental (expressed through prototyping). There are two modes of operation in approaches to design: analytical and experimental, which are used by designers to reduce complexity and uncertainty respectively. Matthiassen & Stage's (1992) *'principle of limited reduction'* states:

> " Relying on an analytical mode of operation to reduce complexity introduces new sources of uncertainty requiring experimental countermeasures. Correspondingly, relying on an experimental mode of operation to reduce uncertainty introduces new sources of complexity requiring analytical countermeasures." (Matthiassen & Stage, 1992, page 173)

From the above discussion of problem-solving conceptualizations, it can be seen that there are radically different ways of 'seeing' the design 'problem' (or problem situation) and each way has different methods for reducing problem uncertainty and complexity. Figure 5 maps the perceived degree of complexity and uncertainty with respect to the product of design (the design 'problem'). This framework permits an analysis of how a changing perception of the design *product* affects the *process* of design.

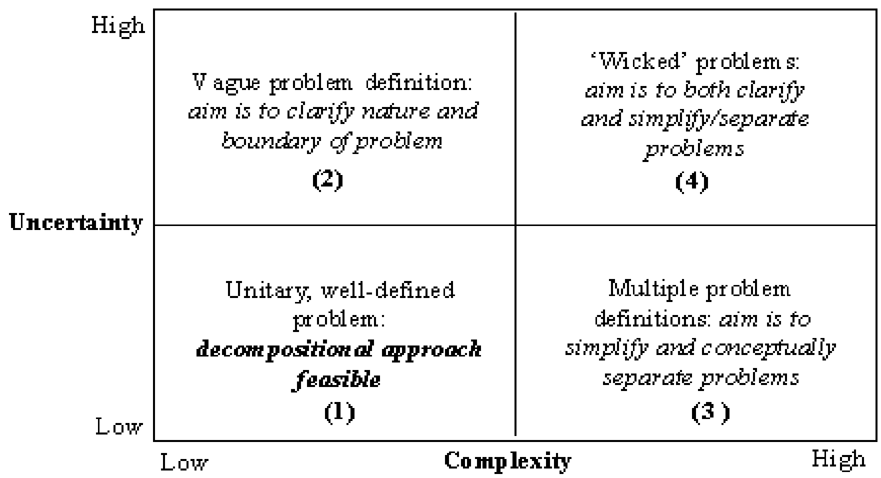

Figure 5. Uncertainty Vs. Complexity In Information System Design

It should be stressed that this framework represents individuals' *perceptions* of the design problem, which may predicate the design approach chosen. Quartile 1 is the 'holy grail' of traditional IS design. Low complexity, coupled with low uncertainty as to the design problem make solution requirements specification and decomposition straightforward: this is the basis of the waterfall model. At the opposite end of the design spectrum, high uncertainty, coupled with high levels of complexity in the perceived problem (quartile 4) indicate a 'wicked' problem. The designer's natural reaction to this perception may be to move the problem into one of the other quartiles of the matrix by applying a reductionist design approach, aimed at reducing either problem uncertainty or problem complexity, in order to make the design problem more manageable (permitting the designer to reduce or simplify the problem). ***Each of the four perspectives discussed above deals with complexity and uncertainty in different ways.***

1. The *rational perspective* (design as functional analysis) reduces complexity by applying scientific reductionism to the problem so that the problem as defined is well-structured and uncontentious. Uncertainty is not an issue as the problem is seen as unitary and well-structured.

2. The *bounded-rationality perspective* (design as problem-solving) sees the problem-solver as reducing an unstructured problem to a set of well-structured sub-problems; this process reduces complexity, but leaves uncertainty to be dealt with by the application of scientific reductionism in defining sub-problems.

3. The *systemic perspective* (design as problem-setting) sees complexity as a necessary (and therefore not undesirable) facet of organizational problem situations. Uncertainty is reduced through the negotiated definition of problem scope (the "system boundary") and achieving consensus on system definition; complexity is managed, rather than being reduced, by the shared learning which accrues through the joint exploration of shared "system" definitions.

4. The *emergent perspective* (design as evolutionary learning) sees both complexity and uncertainty as natural and not necessarily reducible. Complexity is dealt with, to some extent, by the learning-through-doing which accrues from "reflective action" (Schön, 1983) but uncertainty can only be managed in the short-term, by the definition of intermediate goals

> which are accepted as partial in nature. In the longer-term, uncertainty may be viewed as productive, as it leads the individual to engage in reflective learning.

The situated, emergent nature of IS design can be viewed as one of the four perspectives presented in this section. The critical processes of design are concerned with the more or less objective analysis of solutions to design problems, for the first two perspectives of design as functional analysis and as problem-solving. The perspective of design as problem-setting sees the exploration and definition of suitable design goals and boundaries as critical. But the situated, emergent perspective moves away from the idea that problems or design boundaries may be defined in advance: design is seen as a continuing process of defining, exploring and adapting target system goals through the reflective action involved in design itself. The critical processes of design are thus concerned with reflection, learning, negotiation and adaptation, for this perspective.

These perspectives are, to some extent, incommensurable. Although some ideas are held in common across perspectives, each has arisen as a response to perceived inadequacies in design practice and represents a paradigm held by designers. It is likely that design methods based upon a situated, emergent paradigm would not radically affect the approach to problem-solving taken by those operating under one of the other perspectives.

A major problem with all of the above perspectives is that design is viewed as pertaining to the individual. Even the systemic perspective does not concern itself with how a solution is achieved by a group of designers, acting in concert. The nature of design cognition within an organizational IS *development team* requires to be addressed by *applicable* theories of problem-solving in design.

Design As Social Construction

Situated Learning and Design Practice: The Sharing Of Sociocultural Knowledge

The sociological perspective of design as the social construction of technology (Bjiker et al., 1987; Mackenzie & Wajcman, 1985) has shaped the development of much recent theory of design in context. Design is seen as a socially-shared activity (Brown & Duguid, 1992; Lave, 1991) and learning is seen as central to situated design (Brown & Duguid, 1992; Lave, 1993; Lave & Wenger, 1991; Star, 1992). The importance of external artifacts and representations in clarifying design goals is stressed by empirical studies of design interaction (Flor & Hutchins, 1991; Norman, 1986, 1988, 1991; Star, 1992).

The situated learning perspective is grounded in the work of Lave (1988, 1991; Lave & Wenger, 1991). Lave’s writings are concerned with the process of problem formulation and skill acquisition: she is concerned that cognitive theories of learning in the literature are inadequate as they suggest that all knowledge can be written down in symbolic models. The question of what constitutes cultural knowledge and how such knowledge is communicated and learned, through "legitimate peripheral participation", through which individuals are educated in the normative practices of a sociocultural group is explored in these writings. This concept is linked with that of situated action (Suchman, 1987), which claims that psychological models, in terms of beliefs, goals, schemas, inferences, strategies etc., describe and explain patterns of behavior of an agent in an environment, not processes of the brain. Lave & Wenger (1991) stress the centrality of situated learning to "communities of practice". To master knowledge and skill legitimately, newcomers must "move towards full participation in the sociocultural practices of a community", as in an apprenticeship:

> " Viewing learning as legitimate peripheral participation means that learning is not merely a condition for membership, but is itself an evolving form of membership. We conceive of identities as long-term, living relations between persons and their place and participation in communities of practice. Thus identity, knowing and social membership entail one another." Lave & Wenger (1991), page 53

The importance of normative practice is also recognized by Rosenbrock (1981), who sees this as a constraint upon social change in technical design: while technical professionals adopt existing sociocultural value systems, human-centered methods and approaches to design can have little impact. But this perspective ignores the dynamic nature of sociocultural systems: social groups are not static, but are affected by their changing memberships.

In empirical studies, Curtis et al. (1988) stress the centrality of group learning processes and the critical role played by an ‘expert designer’ in communicating application domain knowledge to other team members. Expertise, according the Curtis et al. (1988) is not defined by technical knowledge, but by a valued operationalisation of technical knowledge in specific, local application domains. Communication and coordination activities are critical to design group functioning. In a study of small group design meetings, Olson et al. (1992) discovered that design teams spent only 40% of their time on direct discussions of design. The groups spent one-fifth of their time on "pure coordination activities" and *one-third* of their time on "clarification of ideas" - the sharing of expertise among group members. Walz et al. (1993) performed an observational study of design knowledge sharing and acquisition in videotaped design meetings; the authors "were surprised to see how important context-sensitive learning was to the design process". The issue was raised that much information was presented to the team during design meetings but never captured; Walz et al. (1993) do not examine whether this information was important and what prompted information capture.

Formal project documentation and design models - what Flor & Hutchins (1991) refer to as "external, structured representations" may play a significant role in the communication of situated knowledge in design teams. The way in which design information is represented fundamentally affects the way in which knowledge about that design is communicated and conceptualized (Simon, 1988; Winograd & Flores, 1986). Checkland & Scholes (1990) emphasize the centrality of the group learning which accrues from the production of joint models of the target object system during SSM workshops, to effective participation in organizational activity. Olson et al. (1993) observed that small design groups working with and without a group editor (which produced external structured representations of the design) generated more design ideas without the editor, but fewer and better ideas with it, indicating that the representations helped the supporting groups to remain more focussed on the core issues in the emerging design and to capture what was said as they proceeded. Flor & Hutchins (1991) note that external representations are critical in achieving intersubjective understanding; this concept is explored in the next section.

Intersubjectivity And Distributed Cognition

Curtis et al. (1988) conclude that "developing large software systems must be treated, at least in part, as a learning, communication and negotiation process." Designers have to integrate knowledge from several domains before they can function well. They identify the importance of designers with a high level of application domain knowledge: in their studies, these individuals were regarded by team members as "exceptional designers", who were adept at identifying unstated requirements, constraints, or exception conditions, possessed exceptional communication skills. Exceptional designers spent a great deal of their time communicating their vision of the system to other team members, and identified with the performance of their projects to the point where they suffered exceptional personal stress as a result. They dominated the team design process, often in the form of small coalitions, which "co-opted the design process". While these individuals were important for the *depth* of a design study, teams were important for exploring design decisions in *breadth* (ibid.).

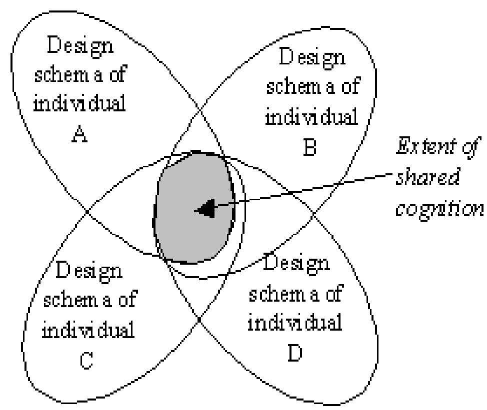

Figure 6. The Concept Of Shared Cognition (adapted from Laukkanen, 1994)

The acquisition of knowledge by design teams involves both *shared* cognition and *distributed* cognition. The concept of shared cognition is illustrated in Figure 6 and represents the extent of intersubjectivity (shared meanings) between organizational actors. Design depends upon intersubjectivity for effective communication between team members to take place (Flor and Hutchins, 1991; Hutchins, 1990, 1991, 1995; Orlikowski & Gash, 1994; Star, 1989). Technical system designers, "successful in sharing plans and goals, create an environment in which efficient communication can occur" (Flor and Hutchins, 1991). Orlikowski & Gash (1994), in a hermeneutic analysis of different interest groups' assumptions, knowledge and expectations of a new groupware technology, refer to intersubjectively-held mental models as "shared technological frames":

> " Because technologies are social artifacts, their material form and function will embody their sponsors' and developers' objectives, values, interest and knowledge regarding that technology" *(Orlikowski & Gash, 1994, page 179).*

The importance of intersubjectivity is emphasized by Lanzara (1983), who sees design as a dynamic process of framing and reframing of situations by a transactive process where different actors negotiate "their perspectives, values and (even!) facts". The different metaphors used (*c.f.* Morgan, 1986), at different times or by different actors, to frame the target system lead designers to emphasize differing *objects* through the design methods which they use. He gives the example of two extremes of metaphor in office systems design: the office may be seen as a "machine", in which case the object of design is functions and procedures, or it may be seen as a "community", in which case the object of design is conversations and transactions (Lanzara, 1983).

Little is known about how developers themselves perceive intersubjectivity or frames of reference (Flor and Hutchins, 1991; Orlikowski & Gash, 1994). Jeffries et al. (1981) stress the importance of 'learning by doing': experience enables concepts to be linked on the basis of the utility of keeping the concepts together and provides a coherent executive model of the processes of and of possible appropriate solution forms to design problems: in design teams this becomes a shared, intersubjectively-held design schema, which enables the team to function coherently.

The objects of different actors' frames of reference can be seen as distinct and may be in conflict (Corbett, 1995). Orlikowski and Gash (1994) note that the frames of reference of IT managers and designers affect their decisions to adopt a particular information technology and to implement specific features of that technology; whilst those of business managers and users affect the way in which the technology is interpreted and used in business processes. A critical design activity is therefore that of making stakeholders' frames of reference explicit and subject to debate, in the interests of intersubjectivity.

There is a trade-off between intersubjectivity and the exploration of design alternatives: if group members possess too much common ground, they may communicate more efficiently but there may be less of a tendency to explore alternative courses of action (Flor and Hutchins, 1991; Wilson and Canter, 1993). An experimental study by Rugs & Kaplan (1993) stressed the importance of goal congruence (i.e. intersubjectively-held goals) in group decision-making. As might be expected group (shared) goals facilitated greater *normative* influence upon decision-making (based upon social relations) and task goals facilitated greater *informational* influence (based upon evidence about reality). The implications for design teams are that effective consideration of organizational design requirements is only possible when there is a high degree of divergence between individuals' models, whereas effective synthesis of solutions requires much higher levels of intersubjectivity. Viewed from this perspective, the separation of requirements analysis and design makes sense: in the early stages of design, members of a design-team are likely to hold diverging models of design requirements and goals and thus cannot effectively synthesize solutions; during later stages, levels of intersubjectivity may be higher and so solution synthesis could proceed more effectively. However, one should distinguish between co-operative design (where all design team members are working on a solution for the same subproblem) and coordinated design (where design team members synthesize solutions to different subproblems which are then brought together in a system solution): the former may require high levels of intersubjective understanding, the latter may function through distributed cognition.

Design activities do not just take place at the *individual* cognitive level, but also involve *distributed* cognition (Norman, 1991; Hutchins, 1990, 1991, 1995) in the social processes of negotiating design requirements and joint elicitation of design solutions. Distributed cognition implies interdependency between actors' individual schemas:

> " Distributed cognition is the process whereby individuals who act autonomously within a decision domain make interpretations of their situation and exchange them with others with whom they have interdependencies so that each may act with an understanding of their own situation and that of others." *(Boland et al., 1994, page 457)*.

Hutchins (1990) describes the distributed mental models of the situation accessed by an airplane cockpit crew in plotting a course when control equipment broke down. No one actor held a complete model of the situation, but individual actors held both partial models of the solution and a process model which enabled them to coordinate other actors' partial models to reach a complete solution. Hutchins (1991) studied how the social organization of distributed cognition affects the cognitive properties of groups in a study of how communities arrived at shared versus differing understandings. He concluded that cognition in this type of situation is shared among agents in organizationally-prescribed roles and also among the artifacts that they use, such as work-procedures, charts, plans and routines for route-calculation - i.e. that models of how a situation may be handled are embodied in the artifacts used to expedite its handling. This echoes the work of the actor-network theorists (e.g. Callon, 1991; Latour, 1987; ,Law, 1992), in treating technological artifacts as 'non-human actors' in the analysis of the 'web' of distributed interactions in organizational decision-making.

Star (1989) addresses the process of combining evidence from different perspectives, in terms of how decision-group participants decide that sufficient, reliable, and fair amounts of evidence have been collected. Star discusses two studies, one of two different groups of physiologists and one of two different groups of biologists, making the following observations:

- Different groups can co-operate without having good models of each other. They can successfully work together while employing different units of analysis, methods of aggregating data, and different abstractions of data

- Different groups can co-operate although they have different goals, time horizons, and audiences to satisfy

- Coordination activities are supported by creating "boundary objects" which can be adapted locally to needs and constraints while maintaining a global identity.

The concept of boundary object is also raised by Norman (1992) who refers to cognitive aids used by commercial airplane crews. Individuals created artifacts to better understand the state of affairs in the cockpit, such as metal or plastic tabs that pilots move around the outside of the airspeed indicator to help remember critical settings. Crews used checklists to provide a mechanism for shared understanding and group memory. A member of the crew reads the list while others perform appropriate operations or checks - using this mechanism, the entire crew is informed about the state of the aircraft and a record of actions performed and the current state of the aircraft is kept. Designers in the study by Flor & Hutchins (1991) were observed to use external structured representations as a means of sharing knowledge.

Empirical evidence as to how design is managed in groups is inconclusive and scarce. There is evidence to show that intersubjective understandings are key to group design (Flor & Hutchins, 1991) yet there is evidence from studies of non-design group processes which involve coordinated action to indicate that *distributed* understanding - actors holding only partial mental models which are interdependent upon other actors' partial mental models for effective action - is also critical. There is some evidence to show that design is a collective *learning* experience: information systems designers jointly develop an intersubjective model of the system on which they base their design assumptions (Curtis et al., 1988; Flor and Hutchins, 1991; Reynolds & Wastell, 1996; Walz et al., 1993. But here is little evidence of how intersubjectivity and distributed cognition are achieved and maintained in design groups: what processes and criteria are involved in determining if information is significant and in sharing or emphasizing significant information? The concept of distributed cognition implies that design groups adopt satisficing, rather than optimizing behavior in design: individually, each team member does not have the cognitive capacity to understand the whole of a complex system, but individuals construct partial models of the problem situation, expecting that a co-operative design outcome will result from coherent group coordination in design.

So from an individual design focus of imposing structure upon a problem situation to facilitate convergence between a design problem-definition and potential partial solutions, we have reached a group design focus of coordinating and sharing partial design solutions. Lave (1991) suggests that the process of socially shared cognition should not be seen as ending in the internalization of knowledge by individuals, but as a process of becoming a member of a "community of sustained practice". Design groups need to maintain intersubjectively-held mental models of design goals and process, if they are to function effectively, yet each group-member may only hold a part of the knowledge and understanding necessary for design to take place. A group of individuals can pool their partial models to perform design activities through the mechanism of distributed cognition. This may be achieved through the shared meanings attached to artifacts used in common by a group - "boundary objects" (Star, 1989; Norman, 1992) - but there is only one study (Flor & Hutchins, 1995) which examines how such distributed models are maintained and this focussed on the extent to which external representations of the design were shared in a single experiment involving two programmers. Whilst some studies examine shared knowledge and learning in IS design (e.g. Curtis et al., 1988; Reynolds & Wastell, 1996; Walz et al., 1993), they do not examine how individuals frames of reference contribute to group perspectives of the design and how these group perspectives are constructed and maintained. The creation of ‘communities of practice’ (Lave & Wenger, 1991) is critical to coordinated design processes, but there are few studies of the mechanisms for achieving this in information system design. As software tools and methods are usually designed for individual problem-solving, they do not support those design processes which emerge from the social behavior of the development team or the organizational behavior of the company: we need research into the nature of group design processes to inform the provision of design tools.

Conclusions

This paper presented and compared two competing worldviews of design: design as organizational problem-solving and design as social cognition. Differences between the two worldviews are summarized in Table 4.

	Problem-solving	**Social Construction**
Individual	Problem definition and goal-determination (planning vs. emergent action)	Situated learning and constructing mental models of problems and solutions
Group	Deriving and validating consensus models	Learning and constructing joint models
Organization	Stakeholder negotiation of political design objectives, scope & strategy	Social construction of meaning and legitimacy; alignment of diverse interests
Perspective (re: the individual)	Exterior (reflected through objective models of a design, which exist separately from the designer)	Relationship between interior and exterior (reflected through external representations of mental models)
View of Agency	Objective: Plans determine outcomes, to a greater or lesser extent.	Constructionist: agents socially construct their world, or Interactionist: Agent & their world co-constitute each other.
Representative Metaphors	"Surfacing" objectives; Design "rationale"; A "common vision"	"Framing" a design; "Constructing" a model; "External representations"
Management Focus	Achieving consensus and coordinating coherent action	Achieving intersubjectivity and collaboration in distributed cognition

Table 4: Comparison Of Two Worldviews Underlying Organizational IS Design

The "problem-solving" perspective takes as axiomatic that design is (more or less) objective: an individual explores a context (i.e. a problem space), to arrive at a design "solution" which is determined by that context. The individual "defines" a system solution to the perceived problem. The objective (external) nature of the problem context and the defined artifact or system of work activities is not challenged in any of the perspectives which are encompassed by this paradigm.

The social construction perspective is more radical in that it views design as the joint construction of *context* as well as *form* (of the information system or technological artifact). By engaging in normative sociocultural practices, designer construct a jointly-held view of the world (design context) which enables them to work cohesively in constructing an artifact or system which engages this worldview. The nature of this engagement has been touched upon in the work of Checkland (1981, Checkland & Scholes, 1990), who includes consideration of the Weltanschauung (worldview) of actors involved in a "problem situation" as part of his systemic approach to organizational problem-exploration, but who fails to engage with the social nature of this worldview.

It can be argued that both paradigms have a place in organizational information system design practice: they are complementary. On the one hand, designers must operate in a design-space which is ontologically viewed as having an existence independent of the design-team. On the other hand, designers must recognize that, through their group discussions and models, they construct a "problem context" which is very different from the context recognized by other actors in the organization. Any effective method of managing organizational IS design must encompass both of these views of action.

References

Ackoff, R.L. (1974) *Redesigning The Future*, Wiley

Alexander, C. (1964) *Notes On The Synthesis Of Form*, McGraw Hill, New York

Ball, L.J. & Ormerod, T.C. (1995) Structured and opportunistic processing in design: a critical discussion', Int. Journal of Human-Computer Interaction, Vol. 43, pp 131-151

Bijker, W.E., Hughes, T.P. and Pinch, T.J. (1987) (Eds.) *The Social Construction of Technological Systems, New Directions in the Sociology and History of Technology*, MIT Press, Cambridge, MA

Boland, R. and Day, W.F. (1989), The experience of systems design: a hermeneutic of organisational action, *Scandinavian Journal of Management*, 5,2 87-104

Boland, R J, Tenkasi, R V, Te'eni, D (1994) Designing Information Technology to Support Distributed Cognition, *Organization Science*, Vol 5, No 3, pp 456-475

Brown, J.S. & Duguid, P. (1992) 'Enacting Design for the Workplace' in P.S. Adler & T.A. Winograd (Eds.) *Usability: Turning Technologies Into Tools*, ACM Press, New York, NY

Burrell, G. (1983) 'Systems Thinking, Systems Practice: A Review', *Journal Of Applied Systems Analysis*, 10.

Callon, M. (1991) ‘Techno-Economic Networks and Irreversibility’, in J. Law (Ed.) *A Sociology of Monsters. Essays on Power, Technology and Domination*, Routledge, London, UK

Ciborra, C.U. (1987) ‘Research Agenda For A Transaction Costs Approach To Information Systems’, in R.J. Boland, and R.A. Hirschheim (eds.) *Critical Issues In Information Systems Research*, Wiley.

Clegg, C (1994) Psychology And Information Technology: The Study Of Cognition In Organisations, British Journal of Psychology, 85, pp 449-477

Corbett, J.M. (1995) ‘IT And Work Design: The Negotiation Of Expertise’ in Scarborough, H. (Ed) *The Management of Expertise,* London, MacMillan.

Corbett, J.M., Rasmussen, L.B. & Rauner, F. (1991) *Crossing the Border: The social and engineering design of computer integrated manufacturing systems*, Springer-Verlag, London

Curtis, B., Krasner, H. and Iscoe, N. (1988) ‘A Field Study Of The Software Design Process For Large Systems’, *Communications of the ACM*, Nov 1988, Vol 31, No 11, pp 1268-1287

De Marco, T. (1979) *Structured Analysis and System Specification*, Yourdon Inc.

Dorst, K. and Dijkhuis, J. (1996) ‘Comparing paradigms for describing design activity’, in N. Cross, H. Christiaans & K. Dorst (Eds.) *Analysing Design Activity*, John Wiley & Sons, Chichester, UK

Flood, R L (1995) ‘What is happening when you problem-solve? A Critical Systems Perspective’, Systems Practice, Vol. 8 No 2, pp 215-222

Flor, N.V. & Hutchins, E.L. (1991) ‘Analyzing distributed cognition in software teams: a case study of team programming during perfective software maintenance’, in J. Koenemann-Belliveau, T. Moher & S. Robertson (Eds.) *Empirical Studies of Programmers - Fourth Workshop*, Norwood, NJ: Ablex

Galliers, R.D. (1987) An Approach To Information Needs Analysis, in Galliers, R.D. [Ed.], Information Analysis, Selected Readings, Addison-Wesley

Galliers, R.D. (1993a) SSM??

Gane, C. & Sarsen, T (1979) *Structured Systems Analysis: Tools and Techniques*, Prentice-Hall, New Jersey

Gasser, L. (1986) The integration of computing and routine work, *ACM Transactions On Office Information Systems*, Vol. 4, pp 205-252

Gilhooley, K.J. (1989) ‘Human and Machine Problem-Solving, A comparative Overview’ in K.J. Gilhooley (Ed.) *Human and Machine Problem-Solving*, Plenum Press, New York

Guindon, R. (1990a) ‘Designing the design process: Exploiting opportunistic thoughts’, *Human-Computer Interaction*, No. 5

Guindon, R. (1990b) ‘Knowledge Exploited By Experts During Software System Design’, *International Journal of Man-Machine Studies*, Vol. 33, October 1990, pp 279-304

Hutchins, E. (1990), ‘The Technology of Team Navigation’, in J. Galagher, Robert E. Kraut, and Carmen Egido (eds.) *Intellectual Teamwork: Social and Technological Foundations of Cooperative Work*, Hillsdale, NJ: Lawrence Erlbaum Associates, Publishers. pp. 191-220.

Hutchins, E. (1991), ‘The Social Organization of Distributed Cognition’, in Lauren B. Resnick, John M. Levine, and Stephanie D. Teasley (eds.) *Perspectives on Socially Shared Cognition*, Washington, DC: American Psychological Association. pp. 283-307.

Hutchins, E. (1995) *Cognition in the Wild*, MIT Press, Bradford

Jeffries, R., Turner, A.A., Polson, P.G. & Atwood, M.E. (1981) ‘The Processes Involved In Designing Software’, in J.R. Anderson (ed.) *Cognitive Skills And Their Acquisition*, Lawrence Erlbaum Associates, Hillsdale, New Jersey

Khushalani, A., Smith, R. & Howard, S. (1994) ‘What Happens When Designers Don’t Play By The Rules: Towards a Model of Opportunistic Behaviour In Design’, Australian Journal of Information Systems, May 1994, pp 13-31

Lanzara, G.F. (1983) ‘The Design Process: Frames, Metaphors And Games’, *in* U. Briefs, C. Ciborra, L. Schneider (eds.) *Systems Design For, With and By The Users*, North-Holland Publishing Company

Latour, B. (1987) *Science in Action*, Harvard University Press, Cambridge, MA

Lave, J. (1991) ‘Situating Learning In Communities of Practice’ in L.B. Resnick, J.M. Levine, S.D. Teasley (Eds.) *Perspectives on Socially Shared Cognition*, American Psychological Association, Washington DC, pp 63-82

Lave, J. & Wenger, E. (1991) *Situated Learning: Legitimate Peripheral Participation*, Cambridge University Press, Cambridge, UK

Law, J. (1992) ‘Notes On The Theory of the Actor-Network: Ordering, strategy and Heterogeneity’ *Systems Practice*, Vol. 5, No. 4, pp 379-393

Law, J. & Callon, M. (1992) ‘The Life and Death of an Aircraft: A Network Analysis of Technical Change’ in W.E. Bijker and J. Law (Eds.) *Shaping Technology/Building Society: Studies In Sociotechnical Change*, MIT Press, Cambridge, MA

Lawson, B. (1990) *How Designers Think*, The Architectural Press, London, UK (Second Edition).

Mackenzie, D.A. & Wajcman, J. (1985) 'Introduction' to Mackenzie, D.A. & Wajcman, J. (eds) *The Social Shaping Of Technology*, Open University Press, Milton Keynes

Malhotra, A., Thomas, J., Carroll, J. & Miller, L. (1980) 'Cognitive Processes In Design', *International Journal of Man-Machine Studies*, 12, pp 119-140

Matthiassen, L. & Stage, J. (1992) 'The Principle of Limited Reductionism In Software Design', *Information Technology & People*, Vol. 6, No. 2, pp 171-185

Mayer, R.E. (1989) 'Human Nonadversary Problem-Solving' in K.J. Gilhooley (Ed.) *Human and Machine Problem-Solving*, Plenum Press, New York

Methlie, L.B. (1980) 'Systems Requirements Analysis - Methods And Models', in H.C. Lucas, F.F. Land, J.J. Lincoln, K. Supper (Eds.) *The Information Systems Environment*, Proceedings of IFIP WG8.2, Bonn 1979, North-Holland Publishing Company

Mintzberg, H. & Waters, J.H. (1985) 'Of Strategies, Deliberate and Emergent', Strategic Management Journal, vol. 6, pp 257-72)

Morgan, G. (1986) *Images Of Organization*, Sage Publications, California

Newell, A. & Simon, H.A. (1972) *Human Problem-Solving*, Prentice Hall

Norman, D.A. (1986) 'Cognitive Engineering', *in* D.A. Norman & S.W Draper (Eds.) *User-Centred System Design: New Perspectives on Human-Computer Interaction*, Erlbaum Associates, Hillsdale, New Jersey

Norman, D.A. (1988) *The Psychology of Everyday Things*, Basic Books, Harper-Collins, USA (Previously published as: The Design of Everyday Things. 1990. New York: Doubleday)

Norman, D.A. (1991) 'Cognitive Artifacts', in J.M. Carroll (Ed.) *Designing Interaction: Psychology At The Human-Computer Interface*, Cambridge University Press, UK, pp 17-38

Norman, D. A. (1992) 'Turn Signals are the Facial Expressions of Automobiles', Addison-Wesley Publishing Company, Inc.. Reading, MA

Norman, D. A. (1993) 'Cognition in the Head and in the World: Introduction to the Special Issue on Situated Action', *Cognitive Science*, Vol. 17, No. 1, pp 1-6

Olson, G.M., Olson, J.S., Carter, M.R., Storrosten, M. (1992) 'Small group design meetings: an analysis of collaboration', *Human-Computer Interaction*, Vol. 7, pp 347-374

Orlikowski, W.J. & Gash, D.C. (1994) 'Technological Frames: Making Sense of Information Technology in Organizations', ACM Transactions on Information Systems, April 1994.

Preston, A (1991)??

Reynolds, M. & Wastell, D. (1996) 'Breaking Down The Social Defences: Enabling Learning With Transitional Objects' in *Proceedings of 4th European Conference on Information Systems*, Lisbon, Portugal, July 2-4 1996

Rosenbrock, H.H. (1981) 'Engineers And The Work That People Do', *IEEE Control Systems Magazine*, Vol 1, No.3 pp 4-8 September 1981. *Reprinted* in Paton et. al. (1984) *Organisations: Cases, Issues and Concepts*, Paul Chapman Publishing, London

Rugs, D. & Kaplan, M.F. (1993) 'Effectiveness of informational and normative influences in group decision making depends on the group interactive goal', *British Journal of Social Psychology*, Vol. 32, pp 147-158

Simon, H.A. (1960) *The New Science of Management Decision*, Harper & Row, New York

Simon, H.A. (1973) 'The Structure of Ill-Structured Problems', *Artificial Intelligence*, No. 4, pp 145-180

Simon, H.A. (1981) *Sciences of The Artificial*, Second Edition, MIT Press, Cambridge, MA

Simon, H.A. (1988) 'Scientific Discovery As Problem Solving', Peano Lecture, reprinted in Egidi, M & Marris, R. (Eds.) *Economics, Bounded Rationality and the Cognitive Revolution*, Edward Elgar Publishing Limited, Aldershot, UK (1992)

Star, S. L. (1989), 'The Structure of Ill-Structured Solutions: Boundary Objects and Heterogeneous Distributed Problem Solving', in L. Gasser and M. N. Huhns (eds.) *Distributed Artificial Intelligence*, Vol. II. San Mateo, CA: Morgan Kaufmann Publishers Inc., pp. 37-54.

Star, S. L. (1992), 'The Trojan Door: Organizations, Work and the "Open Black Box" ', *Systems Practice*, Vol. 5, No. 4.

Suchman, L. (1987) *Plans And Situated Action*, Cambridge University Press, MA, USA

Taylor, F.W. (1947) *Scientific Management*, Harper & Row

Truex, D. and Klein, H.K. (1991) 'A Rejection of Structure as a Basis for Information Systems Development', in R.K. Stamper, P. Kerola, R. Lee and K. Lyytinen (eds.) *Collaborative Work, Social Communications and Information Systems*, Elsevier Science Publishers B.V. (North Holland) IFIP 1991

Turner, J.A. (1987) 'Understanding The Elements Of Systems Design', in Boland, R.J. and Hirschheim, R.A. (eds.) *Critical Issues In Information Systems Research*, Wiley.

Veryard (1986)??

Walz, D., Elam, J. & Curtis, B. (1993) 'Inside A Design Team: Knowledge Acquisition, Sharing and Integration', *Communications of the ACM*, October 1993, Vol. 36, no. 10, pp 63 - 77

Winter, M.C., Brown, D.H. & Checkland, P.B. (1995) 'A Role For Soft Systems in Information Systems Development, *European Journal of Information Systems,* 4 pp 130-142

Wood-Harper, A.T. (1990) 'Characteristics of Information Systems Definition Approaches', *in* R.L. Flood, M.C. Jackson, & P. Keys (Eds.) *Systems Prospects: The Next Ten Years of Systems Research,* Plenum Press, New York

Yourdon, E. (1989) *Modern Structured Analysis,* Yourdon Press, Englewood Cliffs, New Jersey

Yourdon, E. (1993) *Yourdon Systems Method: Model-Driven Systems Development,* Yourdon Press, Englewood Cliffs, New Jersey

Yourdon, E., and Constantine, L.L. (1975) *Structured Design,* Yourdon Press, Englewood Cliffs, New Jersey

Emergence in Organizational 'Problem-solving': Theories of Social Cognition

Susan Gasson, the iSchool at Drexel

Please cite this paper as:
Gasson, S. (2006) 'Emergence in Organizational 'Problem-solving': Theories of Social Cognition',
Last updated 10/04/2007 12:09:36 ***Working Paper***, last updated 07/21/2007; retrieved <insert date of retrieval here> . Available from http://www.cis.drexel.edu/faculty/gasson/papers/ProbSolv.html

Abstract

This paper explores theories underlying the design and development of organizational information systems (IS). It summarizes and compares two very different perspectives on IS design and development: design as the (individual) solution of organizational "problems" and design as the (joint) construction of sociocultural artifacts. The different paradigms of operation engendered by these perspectives is discussed in terms of its implications for information system development (ISD) methods and the management of IS-related organizational change.

Introduction

IS design deals with knowledge-sharing across organizational groups and disciplines. For knowledge to be shared effectively across these organizational "boundaries", we need approaches that view organizational problems as "wicked" problems -- problems that are highly interrelated, hard-to-define and negotiated. Stakeholders from multiple groups and disciplines are required to pool their (often incomplete) knowledge of how the organization works, how they think it should be changed and why, and how information technology should support these changed business processes. This is especially complex when you reflect that people from different parts of an organization (for example, accountants vs. product engineers) often view each other as coming from a different planet! So we need different methods for this type of design initiative -- methods that establish a common language for communication and make explicit the rationale underlying different stakeholder requirements, in a way that stakeholders from other disciplines can understand.

Design As The Solution of Organizational Problems

Design in the ISD literature is often regarded in the context of 'problems':

> " design is to be viewed as the process of problem understanding and problem solving with the aim of producing an artifact." *(Khushalani et al., 1994, page 13)*.

The assumption that problems may be clearly understood and defined in the context of IS design is a misapprehension. Curtis et al. (1988) quote a system engineer, from one of their empirical studies of large development teams:

> " Writing code isn't the problem, understanding the problem is the problem." *(ibid., page 1271).*

This perspective of IS design is also reflected in the organizational management literature:

> " We fail more often because we solve the wrong problem, than because we get the wrong solution to the right problem." *(Ackoff, 1974).*

Lanzara (1983) argues that an understanding of the model which underlies problem-solving behavior in design can explain the core problems which constrain the information system design process in organizations. He identifies three models of information system (IS) design:

1. Design as functional analysis. In this model, design goals and criteria for achieving those goals are pre-defined; the process of design is one of *rationally selecting* means for achieving given ends.

2. Design as problem-solving. The design context contains cues which permit the designer to perform an intelligent search by which the designer *learns* about the structures inherent in the situation and to construct an interface between those structures and the external context. The final solution is just one among many that are feasible.

3. Design as problem-setting. This involves a process of collective inquiry and search taking place through transactions and conversations among several actors with mixed interests concerning the problem at hand. What needs to be created is what the problem-solving model takes for granted: an appropriate problem representation which reflects a decisional structure.

While each of the above models can be found underlying both theoretical and empirical studies of IS design, these three models are insufficient to represent a contemporary understanding of design processes. An additional model is therefore added here:

4. Design as evolutionary learning. The notion that structure is inherent in a situation (as in model 2) is rejected: organizational contexts are seen as dynamic and the objective of design is viewed as an evolutionary convergence between problem-understandings and solution-definitions. This process involves reflective action on the part of the individual: i.e. learning-by-doing, where

individuals' courses of action are created and modified by the organizational structures they are acting upon and individuals' actions create and modify organizational structures in turn.

A design context may be characterized by the degree of complexity: the amount of relevant information available in a given situation; and the degree of uncertainty: the availability and reliability of the information available (Matthiassen & Stage, 1992). Each of the models of design is discussed with respect to how it deals with uncertainty and complexity in organizational problem-definition.

Design As Functional Analysis

This model, described as "functional analysis" by Lanzara (1983), is rooted in the scientific management tradition (Taylor, 1947). The rational model of design is based upon a computer information processing model of human cognition (shown in Figure 1), which assumes that all information pertaining to design requirements is available to the designer and that such information can be easily assimilated (Mayer, 1989).

PROBLEM PRESENTATION
--- (apply representational processes)→
PROBLEM REPRESENTATION
--- (apply solution processes)→
PROBLEM SOLUTION

Figure 1. An Information-Processing Model of Problem-Solving (Mayer, 1989)

Psychological models of human problem-solving have been dominated by the metaphor of computer information-processing (Gilhooley, 1989). Representation involves moving from a statement of the problem in the world to an internal encoding of the problem in memory by mentally encoding the given state, goal states, and legal operators for a problem - i.e. by structuring the problem. Solution involves filling in the gap between the given and goal states, by devising and executing a plan for operating on the representation of the problem - i.e. by making a rational choice between alternative courses of action. This model assumes that there is perfect knowledge of design requirements before the problem is structured, as in Alexander's (1964) "synthesis of form".

The functional analysis approach to design involves scientific reductionism (Corbett et al., 1991; Wood-Harper, 1990). A single, technical problem-definition is derived from a 'rational' analysis of organizational goals (Galliers, 1987); this reduces design uncertainty. The removal of organizational and social aspects pertaining to the IS "problem" until the system requirements are defined solely in terms of technical functions reduces problem complexity (Matthiassen & Stage, 1992). While the rational, problem-solving model underlies many of the structured approaches to IS design (e.g. De Marco, 1979; Gane & Sarsen, 1979; Yourdon & Constantine, 1975; Yourdon, 1989, 1993) and has been very influential in forming practitioners' expectations of the process of design, it does not reflect the complexity of problem-solving seen in organizational information system design.

The information-processing perspective uses the 'machine' metaphor to describe the organization: humans may make a rational decision between alternative solutions only if organizational problems are sufficiently structured to be solved by choice between alternative solutions. Mayer (1989) questions four premises of the information-processing model: that humans can pre-determine what course(s) of action are required to reach a given state; that problem representation and solution are independent of each other; that organizational problem-solving can be accomplished by mechanical, algorithmic processes; and that novel problems can be solved by deductive, rather than inductive reasoning. The information-processing perspective is refuted by empirical research, which indicates that designers solve novel problems by generalizing from a similar problem, engaging in random solution attempts or reframing the problem (Lawson, 1990; Mayer, 1989; Malhotra et al., 1980; Turner, 1987).

Design As Problem-Solving

Simon (1960, 1973, 1981; Newell & Simon, 1972) rejected the rational model of problem-solving, with the notion of "bounded rationality". Simon argued that the rational problem-solver is assumed to understand all information relevant to the problem, and to have clear goals and priorities. The concept of bounded rationality accepts that human-beings have cognitive limitations which constrain the amount of information they can absorb and process; the complexity involved in processing and evaluating available information can prevent the individual from selecting the optimal outcome. Individuals also have access to incomplete information about alternative courses of action, which leads to high levels of uncertainty on the part of the individual. Individuals respond to problem uncertainty by developing a simplified model of the real situation: "bounding" the problem until it becomes sufficiently well-defined to be resolved, they then evaluate alternative solutions sequentially until an alternative is discovered which satisfies an implicit set of criteria for a satisfactory solution. The solution reached is not optimal, but *satisficing*, in that it satisfies a minimal, rather than optimal set of solution criteria. (Simon, 1981).

Simon (1973) describes design problems as "ill-structured" problems. Guindon (1990b) presents a framework for distinguishing between well-structured and ill-structured problems; this is given in Table 1.

Well-structured problems	**Ill-structured problems**
Complete and unambiguous specification of problem	Incomplete and ambiguous specification of the problems
Definite criteria to evaluate the solution and mechanizable process for evaluating if a solution is reached	No stopping rule - no definite criteria to evaluate whether a solution is reached
Any knowledge needed by the problem solver can be represented in one or more "problem spaces"	Many sources of knowledge (problem spaces) that cannot be determined in advance and need to be integrated
Enumerable set of operators that can change the initial state into another state and there is at least one problem space in which can be represented initial state, goal state and all intermediate states	No exhaustive, enumerable list of operators to reach a solution and absence of predetermined solution path from initial state to goal state
Examples: Checkers, Tower of Hanoi, Chess, Theorem-Proving	*Examples:* Design (software, architectural), Planning, Management, Document and music composition

Table 1: Some contrasting features between well-structured and ill-structured problems *(Guindon, 1990b)*

In Simon's model of bounded rationality, individuals decompose an ill-structured problem under the control of a mental, executive process that carries out the necessary coordination functions. Additional information, retrieved from long-term memory, converts the original, ill-structured problem into a collection of well-structured problems: i.e. the process involves inductive reasoning, in addition to the 'rational', deductive reasoning assumed by the rational model of functional analysis. The nature of the problem is no longer unitary - as in the functional analysis model of design - but scientific reductionism is still an integral part of the bounded rationality model of design behavior, as the individual simplifies the problem to reduce uncertainty. The ill-structured organizational problem is viewed as reducible to a set of well-structured sub-problems. Inductive abstraction is required to reduce complexity; Simon (1973, 1981) argues that this behavior is far from the 'rational' decomposition of problem requirements assumed by the functional analysis model of design. In a hermeneutic study of design performed by Boland & Day (1989), a system designer was observed to deal with organizational complexity and political conflict by defining the system in a way which excluded organizational and political issues.

Studies of highly-skilled practitioners indicate that they rely on "intuition" (i.e. inductive reasoning) to problem-solving; such individuals are said to be in the autonomous and most advanced stage of knowing (Anderson, 1983). Schön (1983) refers to this application of intuitive reasoning as "reflection-in-action". Expert system designers have been observed to apply "data-driven rules" (Guindon, 1990a) - the extrapolation of empirical solutions for similar technical problems - rather than more effective goal-directed behaviors, as data-driven behavior imposes a lower cognitive cost (Anderson, 1983).

Design As Problem-Setting

Wood and Wood-Harper (1993) argue that the use of information technologies has been dominated by the rationalistic tradition discussed in the two categories above; they suggest that design of 'new' technology requires "a focus on the formulation of the 'problem' rather than merely providing an 'objective' description of the problem" (ibid., page 100). But Guindon (1990b) argues that information system design involves the integration of ***multiple*** knowledge domains: the application domain, software system architecture, computer science, software design methods, etc.. Each of these domains represents a problem-space in which a more or less guided search takes place (depending upon which solution paths look most promising and the previous experience of the designer in this domain). The IS development process should encompass the discovery of new knowledge, in particular the discovery of unstated goals and evaluation criteria. Rittell (1972; Rittell and Webber, 1973) defines organizational problem-situations as "wicked" problems. While the concept of wicked problems is similar to Simon's (1960) concept of ill-structured problems, in Simon's (1960) perspective, ill-structured problems may be structured by the application of suitable decompositional analysis techniques - i.e. they may be ***analyzed*** (even if not rationally, in a way that may be justified on rational grounds) - whereas wicked problems cannot be formulated because of their complexity and their interrelatedness with other organizational problems (Rittel & Webber, 1973); they must be ***framed***: defined in terms of contextual frames of reference. A wicked problem (Rittell 1972; Rittell and Webber, 1973) has the following characteristics:

- It is unique.
- It has no definitive formulation or boundary.

- There are no tests of solution correctness, as there are only ‘better’ or ‘worse’ (as distinct from right or wrong) solutions.
- There are many, often incompatible potential solutions.
- The problem is interrelated with many other problems: it can be seen as a symptom of another problem and its solution will formulate further problems.

Whereas, in the problem-solving model, problems may be objectively bounded and decomposed, solutions to wicked problems require a more subjective approach: Rittell (1972) advocates ‘second-generation design methods’ to replace the rational model of design. These methods should include "designing as an argumentative process", which Rittell sees as "a counterplay of raising issues and dealing with them, which in turn raises new issues and so on". This approach is more related to the third model of design described by Lanzara (1983): design as the search for appropriate problem-definitions as well as solutions.

An area of research which explores how designers explore the problem space is the work on ‘design rationale’. Echoing Rittel’s (1972) advocation of ‘second-generation design methods’ (i.e. involving argumentation and debate), the ‘design rationale’ theorists (e.g. Buckingham Shum et al., 1996; Lewis et al., 1996; Moran & Carroll, 1996) argue that design requirements are often implicit; they can be surfaced most effectively in the course of social interaction and negotiation if represented explicitly and so made open to inspection and negotiation. The design rationale perspective sees design as taking place within a ‘design space’ - a concept borrowed from cognitive psychology (***c.f.*** Anderson, 1981). The exploration of this design space may be expedited by the explicit representation of design criteria and solution alternatives, using design-space analysis techniques such as the issue-based IBIS technique (Rittel, 1972) or the Question-Option-Criteria (QOC) notation (Maclean et al., 1993).

The design rationale approach sees its objective as recording and understanding the basis of decision-making employed in achieving a (usually technical) design solution. The ***problem*** is defined only inasmuch as the requirements of the designed system are understood and debated by design participants. Whilst the design rationale perspective explores the problem through an exploration of design alternatives and sub-problems and so concentrates upon a problem-space exploration which leads to learning about the solution requirements, the nature of the problem situation and its social and organizational context are ill-defined and remain invisible to the processes of design. Design-rationale goes some way towards the negotiation of problem-definitions: by representing problem-definitions, they are laid open to inspection and negotiation, but these representations are detailed and complex, meaning that non-technical design participants are likely to find their construction problematic, and they are static representations, coping poorly with the emergent and dynamic nature of problem exploration. The approach is seen as most valuable in recording design-space decision-making for the future maintenance of technical systems (e.g. the design of an IT user-interface).

The fifth characteristic of wicked problems (a problem’s interrelatedness with other problems) is characteristic of Ackoff’s (1974) ‘messes’, which are described as "a system of problems" (Ackoff, 1974, page 4). These five characteristics typify the central contradiction of information system design: how may multiple, conflicting problem goals, requirements and constraints be merged into a coherent whole which is acceptable and comprehensible to all those affected by the design - the design team, managers, users, clients and other stakeholders.

Design as problem-setting sees design as a process of collective inquiry and search, taking place through transactions and conversations among several actors with mixed interests concerning the problem at hand (Lanzara, 1983). What needs to be created here is what the problem-solving model takes for granted: an appropriate problem representation which reflects a decisional structure (Lanzara, 1983). While the rational, problem-solving perspective of design attempts to reduce uncertainty through problem-structuring and to reduce complexity through scientific reductionism, the systemic inquiry perspective accepts that complexity is a necessary facet of organizational systems; this perspective manages uncertainty through viewing design as holistic, systemic inquiry into the problem context and manages complexity by recognizing that there are multiple potential target object systems which can be made explicit and subject to negotiation by organizational actors (Checkland, 1981).

The resolution of "wicked" problems in design (Rittel, 1972) is concerned with debate and negotiation to achieve consensus and choice among different, target "object systems" (Welke, 1983); the critical processes are concerned with problem-framing. This perspective argues that there are no objectively given object systems, rather people have viewpoints which enable them to perceive object systems; a multiplicity of viewpoints may prevail among the members of a development team (see Figure 2) and among other stakeholders (Checkland, 1981; Hirschheim, 1985; Hirschheim & Klein, 1992; Lyytinen, 1987). Information system design encompasses multiple perspectives of system objectives, which are ill-defined and open to debate and negotiation (e.g. Mumford, 1983; Checkland & Scholes, 1990; Avison & Wood-Harper, 1990); the subjectivity inherent in this model is encompassed in the use of the term "soft systems" (Checkland, 1981; Checkland & Scholes, 1990).

Lyytinen (1987) proposes that the critical processes of a design team are concerned with achieving consensus on which object system(s) are to be operated upon and the form and scope of the target object system(s).

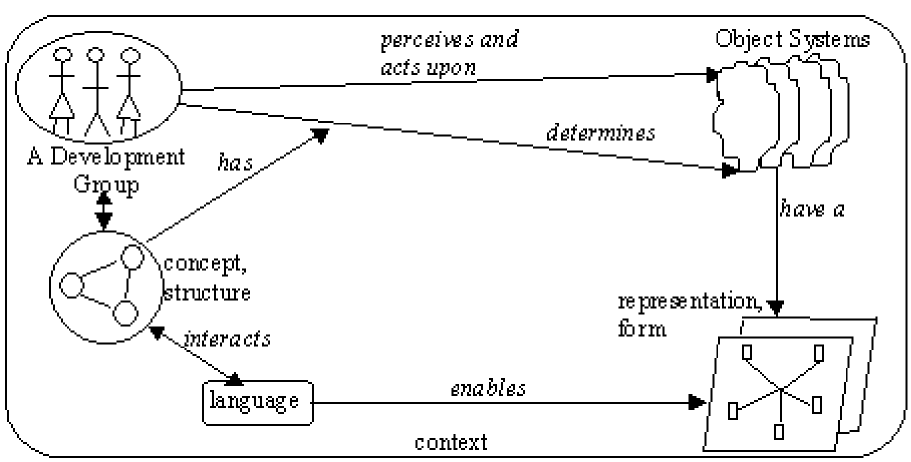

Figure 2. Object Systems In Systems Development (Lyytinen, 1987)

Consensus may mean that perspectives are unitary in nature, reproducing a primary constraint of the previous two approaches to problem-solving. The soft systems perspective is still becoming established: even many writers who have attempted to merge practical approaches to ISD with ‘soft’ approaches speak of a single, ‘primary-task system’ (Wilson, 1984) or the need, early in the IS development process for a "clear statement of system objectives" (Veryard, 1986). Checkland (1980, Checkland & Scholes, 1990) is unclear on this point, but his work has been criticized for privileging the management interest through the search for consensus, which is unrealistic in a political context where management interests dominate (Burrell, 1983). This unitary emphasis can partly be explained by the need for commitments and promises in the negotiated ‘contract’ which represents ISD in practice (Ciborra, 1987) or the need to constrain system requirements in order to meet tight deadlines and resource constraints (Curtis et al., 1988) or because of the individual’s need to reduce problem complexity (Matthiassen & Stage, 1992), but may also reflect the formative influence of rational models of design. Organizational information system design reflects the support of many different tasks and it is the job of the designer - normally a technically trained system analyst - to make sense of the multiplicity of tasks for which support is required and to mediate between the impact of the technical information system being designed and the users of that technology (Boland & Day, 1989). This task involves the resolution of conflicting goals (Methlie, 1980) or multiple constraint satisfaction, defined as "the evolution and testing of part of a design to gradually satisfy its requirements" (Buckingham Shum et al.,1996).

Galliers (1993a) presents an approach based upon SSM, but retaining the plurality of system definitions, while Flood (1995) attempts to deal with the political aspects of systemic design with a "total systems intervention". He suggests that there are three main types of problem-solving methods required for such an intervention: designing, debating and disimprisoning, which are respectively concerned with finding a solution to either efficient processes or effective organizational design, changing people’s beliefs and attitudes, and preventing designs and decisions from becoming ‘prisons’ - i.e. the challenging of received wisdom by questioning whose interests are being served.

For Checkland (1981) soft systems have four main properties: emergence (the exhibition of properties by the whole which are not exhibited by the component parts), hierarchy (entities which can meaningfully be treated as wholes are built up of parts which are themselves wholes, and so on), communication (the transfer of information) and control (the process by which a whole entity retains its identity and performance under changing circumstances). This can be contrasted with the hard systems approach, which sees system properties as being objective, rather than emergent, with communication and control being human interactions with the material (computer-based) ‘system’, rather than properties of the system itself. While soft systems approaches to IS design see IT as the "serving system" to a "served system" of purposeful human-activity (Winter et al., 1995), hard systems approaches see IT as the target object system. However, this view is still static: the soft systems literature views design as being a process of negotiating a consensus on organizational system definitions, which embody structure and persistence. It may also be argued that the whole thrust of the ‘problem’ investigation literature in the field of IS is aimed at structuring problems and constructing structured data (Preston, 1991). An alternative model rejects organizational structure as the basis for design (Truex & Klein, 1991): organizations are seem as emergent and dynamic, with design defined as situated, evolutionary learning.

Situated Design: Dealing With Emergent Goals

This section deals with design as the convergence of problem and solution, as distinct from Lanzara's (1983) last perspective of design as problem-setting. Although design is still viewed as being properly rooted in a process of collective inquiry and search, it is recognized that both problem and solution representations do not reflect an appropriate decisional structure, as required by Lanzara (1983), but are emergent and ill-defined; solutions are no longer optimal for that context, but satisficing. There is evidence in some areas of literature (particularly in the field referred to as social psychology) that our conception of design is changing, with an acknowledgement that design is "situated" in organizational contexts (Gasser, 1986; Suchman, 1987; Lave, 1991; Lave & Wenger, 1991). Star (1992) ascribes this change to "the failure of the rational to account for or to prescribe people's behavior" (Star, 1992, page 398). From problem-solving in a rational sense, the situated action perspective views design as a cyclical process of learning about a situation, then planning short-term, *partial* goals (Suchman, 1987), which emerge from the process of design. The nature of the emerging "problem" becomes more complex and unbounded (and, indeed, unboundable) than that assumed in either the problem-solving perspective or the soft systems perspective. Aspects of a solution are explored in conjunction with aspects of a problem understanding; the designer's understanding of both may change as a result of the process. The 'problem' is thus dynamic and constituted of many, interrelated parts.

The evidence from studies of cognitive design strategy indicate that a further model of problem exploration is required to understand design; even the systemic, problem-investigation model discussed below assumes that it is possible to define organizational problems before appropriate solutions can be formed (although the literature is ambiguous on this point, this assumption does appear to underlie the notion of 'consensus' in the soft systems literature). But an understanding of organizational problems - and appropriate design goals - *emerges* as partial solutions are explored.

Mayer (1989) demonstrated that representation and solution are interactive processes, i.e. the problem representation is continually reformulated during the process of problem solution. Lave & Wenger (1991) argue that design abstraction is situated in the organizational context: abstract representations of a solution are meaningless unless they can be made specific to the concept at hand. This theory is supported by experimental studies of design (in laboratory conditions), which indicate that designers solve novel problems by generalizing from a similar problem, or by reframing the problem to fit partial solutions which are already available to them from their own or colleagues' experience (Lawson, 1990; Mayer, 1989; Malhotra et al., 1980; Turner, 1987). Problem and solution are thus interrelated: this concept is synthesized in the diagram of Figure 3.

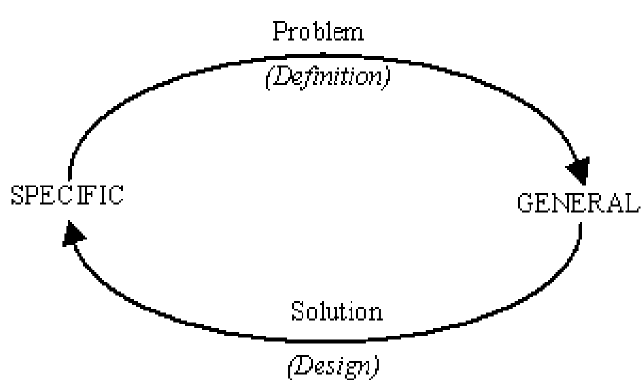

Figure 3. The Inter-related Nature Of Design and Problem Definition

Both Turner (1987) and Malhotra et al. (1980) differentiate IS design from more structured problem-solving by an absence of well-defined goals. In empirical studies of dialogues between designers and clients, Malhotra et al. (1980) observed that designers reframed subproblems to fit an available solution and found that design goals were often partial, implicit and unstated - they were uncovered only when the user stated a system requirement which conflicted with them. They concluded that problem-framing and solution-synthesis were interrelated: problems and solutions converged towards completeness.

So - not only is the problem unclear at the *start* of the process, but the *goals* of the design are also ill-defined in this perspective of problem 'solving' - unlike Checkland's perspective, where one structures the problem situation through defining the goals of a solution. The situated action school of thought considers complication of the problem understanding an aim, rather than complexity-reduction (Boland et al., 1994). Organizations are "organized anarchies" in which people discover analysis and design goals from what they are doing: the processes of bargaining, learning and adaptation (Clegg, 1994).

Turner (1987) argues that "requirements and solutions migrate together towards convergence" and that the process of designing information systems is subjective as well as emergent:

> " Design appears to be more ad hoc and intuitive than the literature would lead us to believe, solutions and problems are interrelated and the generation of solutions is an integral part of problem definition. Problems do not have only one solution; there may be many. Consequently, design completeness and closure cannot be well-defined. There are two categories of design factors: subjective and objective. Objective factors follow from the subjective concepts on which designers model the system. The difficulty in the past is that we have not acknowledged, explicitly, the presence of subjective factors, with the result that, in many cases, objective factors appear to be arbitrary." (Turner, 1987).

Such subjectivity in design problem-investigation is linked with "opportunism" in design (Guindon, 1990a, 1990b; Khushalani et al., 1994). Ball & Ormerod (1995) review the notion of opportunism in system design, which they define as deviation from top-down (decompositional, breadth-first) design approaches and compare opportunistic design with the more structured problem-solving approaches observed in earlier studies of software design. They conclude that much of the structure observed in the early studies of design arose from the more structured nature of the problems set for subjects in experimental situations.

Opportunistic design strategies naturally fit with the prototype or evolutionary systems development approach which permit "learning by doing" (Jeffries et al., 1981). Schön (1983) describes this type of planning through his description of design as "art", which he bases upon the concept of "reflection-in-action"; this concept describes purposeful action which calls on tacit knowledge for its execution. The concept is best described in Schön's (1983) own words:

> " Even when he [the professional practitioner] makes conscious use of research-based theories and techniques, he is dependent on tacit recognitions, judgements and skillful performances." (Schön, 1983, page 50).

This perspective is contrasted with Simon's (1973) 'rational' problem-solving model by Dorst and Dijkhuis (1996), in Table 2. A critical feature of this perspective is that *each design problem is viewed as* *unique*: solutions cannot be analyzed, only inductively synthesized from the social constructions of designers.

Item	'Simon'	'Schön'
designer	= information processor in an objective reality	= person constructing his/her reality
design problem	= ill-defined, unstructured	= essentially unique
design process	= a rational search process	= a reflective conversation with the situation
design knowledge	= knowledge of design procedures and 'scientific' laws	= the artistry of design: when to apply which procedure/piece of knowledge
example/model	= optimization theory, the natural sciences	= art/the social sciences

Table 2: The rational problem-solving paradigm and the reflection-in-action paradigm contrasted *(Dorst and Dijkhuis, 1996)*

The critical processes of design thus become the exploration, representation, sharing and *evolution* of partial, emergent design goals and the inductive assessment of when a satisficing solution has been reached. This perspective is echoed in the work on modeling design rationale, where the importance of generating and recording subproblems - also referred to as microproblems (Lewis et al., 1996) - is a central concern in making design decision-making visible.

The focus is no longer on the individual designer as *decision-maker*, but on the individual as "*conversation-maker*" (Boland et al., 1994), both through reflective action and through interaction with other stakeholders in the design. Human-beings do not plan actions which are followed through without reflection, but are guided by partial plans which are locally contingent upon the context of activities and material conditions involved in the problem situation (Suchman, 1987). Design problems and partial, ill-defined design goals emerge from the processes of engaging in design activity (Hutchins, 1990, 1991, 1995). This concept can be assimilated with the concept of emergent strategy proposed by Mintzberg and Waters (1985): design problem and goal conceptualizations emerge or are discarded continually through the duration of a design.

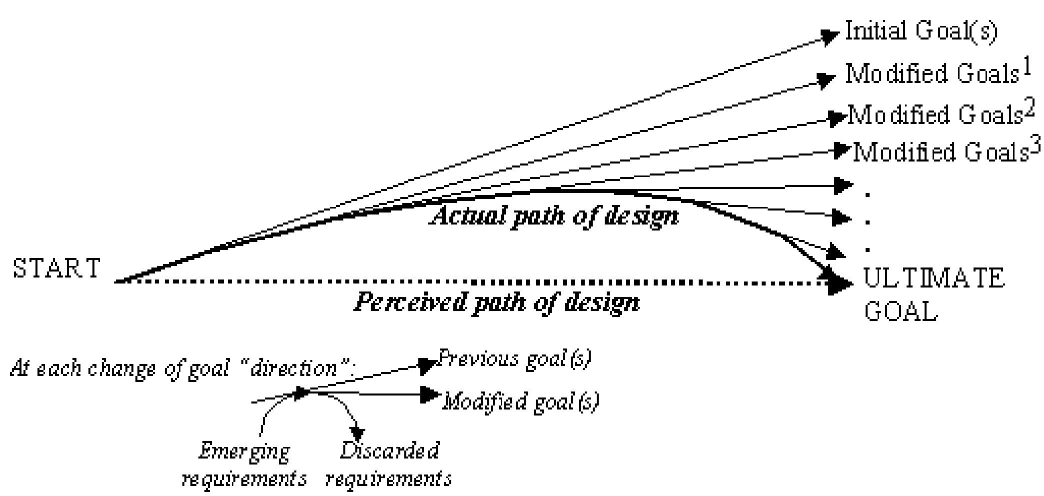

Figure 4. The Implications Of Emergent Design

The implications of emergent design are illustrated in Figure 4. Goals are constantly evolving with an understanding of the design and the actual path of design is much more complex (and longer) than that perceived by actors external to the design process, who only see the start and end points of the design. This model may explain why timescales always ‘slip’ in IS development projects - a common comment from those not involved in such projects is "why did it take you so long?". A critical process of design must therefore be the management of external perceptions of the design process, particularly those of the "global network" (Law & Callon, 1992) - the informal network of influential decision-makers affected by, and indirectly attached to a design project.

Discussion Of The Critical Processes Of Design

A synthesis of the four organizational ‘problem-solving’ perspectives discussed above is given in Table 3. It can be seen from this table that the critical issues of design differ, depending upon the perspective of problem-solving which is adopted.

Problem-solving perspective	**Prescribed process for problem resolution**	**Metaphor for organization**	**Assumed nature of problem**	**Nature of problem-exploration**	**Critical processes of design**
Rational (functional analysis)	Decision between all alternative solutions (parallel analysis)	Machine	Unitary, well-structured	Problem solving	Assessing alternative solutions for optimality
Bounded rationality (problem-solving)	Sequential analysis of solution alternatives	Brain	Ill-structured: reducible to well-structured sub-problems	Problem structuring	Assess alternative solutions against minimal, satisficing set of criteria
Systemic (problem-setting)	Exploration of problem situation; achieving consensus on desirable & feasible action	Web	Multiple, inter-related & socially constructed & static	Collective inquiry and problem-search	Shared investigation of and learning about a problem situation; Structuring problems to explore solutions
Emergent (situated design)	Inductive convergence of emergent problem & solution definitions	Organism	Multiple, inter-related, socially constructed & emergent/dynamic	Reflective action: learning-by-doing	Discovering partial, dynamic goals and solutions through experimentation; managing influential stakeholder perceptions

Table 3: Four Perspectives On The Resolution Of Organizational Problems

According to Matthiassen & Stage (1992) the basic characteristics of a design situation may be described in terms of their degree of uncertainty and their degree of complexity, while approaches to design may be characterized as analytical (expressed through requirements specification) or experimental (expressed through prototyping). There are two modes of operation in approaches to design: analytical and experimental, which are used by designers to reduce complexity and uncertainty respectively. Matthiassen & Stage's (1992) *‘principle of limited reduction’* states:

> " Relying on an analytical mode of operation to reduce complexity introduces new sources of uncertainty requiring experimental countermeasures. Correspondingly, relying on an experimental mode of operation to

reduce uncertainty introduces new sources of complexity requiring analytical countermeasures." (Matthiassen & Stage, 1992, page 173)

From the above discussion of problem-solving conceptualizations, it can be seen that there are radically different ways of 'seeing' the design 'problem' (or problem situation) and each way has different methods for reducing problem uncertainty and complexity. Figure 5 maps the perceived degree of complexity and uncertainty with respect to the product of design (the design 'problem'). This framework permits an analysis of how a changing perception of the design *product* affects the *process* of design.

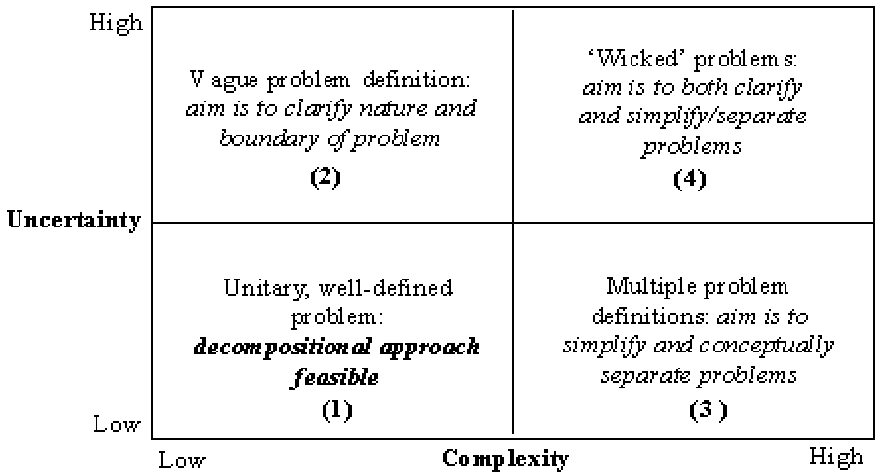

Figure 5. Uncertainty Vs. Complexity In Information System Design

It should be stressed that this framework represents individuals' *perceptions* of the design problem, which may predicate the design approach chosen. Quartile 1 is the 'holy grail' of traditional IS design. Low complexity, coupled with low uncertainty as to the design problem make solution requirements specification and decomposition straightforward: this is the basis of the waterfall model. At the opposite end of the design spectrum, high uncertainty, coupled with high levels of complexity in the perceived problem (quartile 4) indicate a 'wicked' problem. The designer's natural reaction to this perception may be to move the problem into one of the other quartiles of the matrix by applying a reductionist design approach, aimed at reducing either problem uncertainty or problem complexity, in order to make the design problem more manageable (permitting the designer to reduce or simplify the problem). ***Each of the four perspectives discussed above deals with complexity and uncertainty in different ways.***

1. The *rational perspective* (design as functional analysis) reduces complexity by applying scientific reductionism to the problem so that the problem as defined is well-structured and uncontentious. Uncertainty is not an issue as the problem is seen as unitary and well-structured.

2. The *bounded-rationality perspective* (design as problem-solving) sees the problem-solver as reducing an unstructured problem to a set of well-structured sub-problems; this process reduces complexity, but leaves uncertainty to be dealt with by the application of scientific reductionism in defining sub-problems.

3. The *systemic perspective* (design as problem-setting) sees complexity as a necessary (and therefore not undesirable) facet of organizational problem situations. Uncertainty is reduced

through the negotiated definition of problem scope (the "system boundary") and achieving consensus on system definition; complexity is managed, rather than being reduced, by the shared learning which accrues through the joint exploration of shared "system" definitions.

4. The *emergent perspective* (design as evolutionary learning) sees both complexity and uncertainty as natural and not necessarily reducible. Complexity is dealt with, to some extent, by the learning-through-doing which accrues from "reflective action" (Schön, 1983) but uncertainty can only be managed in the short-term, by the definition of intermediate goals which are accepted as partial in nature. In the longer-term, uncertainty may be viewed as productive, as it leads the individual to engage in reflective learning.

The situated, emergent nature of IS design can be viewed as one of the four perspectives presented in this section. The critical processes of design are concerned with the more or less objective analysis of solutions to design problems, for the first two perspectives of design as functional analysis and as problem-solving. The perspective of design as problem-setting sees the exploration and definition of suitable design goals and boundaries as critical. But the situated, emergent perspective moves away from the idea that problems or design boundaries may be defined in advance: design is seen as a continuing process of defining, exploring and adapting target system goals through the reflective action involved in design itself. The critical processes of design are thus concerned with reflection, learning, negotiation and adaptation, for this perspective.

These perspectives are, to some extent, incommensurable. Although some ideas are held in common across perspectives, each has arisen as a response to perceived inadequacies in design practice and represents a paradigm held by designers. It is likely that design methods based upon a situated, emergent paradigm would not radically affect the approach to problem-solving taken by those operating under one of the other perspectives.

A major problem with all of the above perspectives is that design is viewed as pertaining to the individual. Even the systemic perspective does not concern itself with how a solution is achieved by a group of designers, acting in concert. The nature of design cognition within an organizational IS *development team* requires to be addressed by *applicable* theories of problem-solving in design.

Design As Social Construction

Situated Learning and Design Practice: The Sharing Of Sociocultural Knowledge

The sociological perspective of design as the social construction of technology (Bjiker et al., 1987; Mackenzie & Wajcman, 1985) has shaped the development of much recent theory of design in context. Design is seen as a socially-shared activity (Brown & Duguid, 1992; Lave, 1991) and learning is seen as central to situated design (Brown & Duguid, 1992; Lave, 1993; Lave & Wenger, 1991; Star, 1992). The importance of external artifacts and representations in clarifying design goals is stressed by empirical studies of design interaction (Flor & Hutchins, 1991; Norman, 1986, 1988, 1991; Star, 1992).

The situated learning perspective is grounded in the work of Lave (1988, 1991; Lave & Wenger, 1991). Lave's writings are concerned with the process of problem formulation and skill acquisition: she is concerned that cognitive theories of learning in the literature are inadequate as they suggest that all knowledge can be written down in symbolic models. The question of what constitutes cultural knowledge and how such knowledge is communicated and learned, through "legitimate peripheral participation", through which individuals are educated in the normative practices of a sociocultural group is explored in these writings. This concept is linked with that of situated action (Suchman, 1987), which claims that psychological models, in terms of beliefs, goals, schemas, inferences, strategies etc., describe and explain patterns of behavior of an agent in an environment, not processes of the brain. Lave & Wenger (1991) stress the centrality of situated learning to "communities of practice". To master knowledge and skill legitimately, newcomers must "move towards full participation in the sociocultural practices of a community", as in an apprenticeship:

> " Viewing learning as legitimate peripheral participation means that learning is not merely a condition for membership, but is itself an evolving form of membership. We conceive of identities as long-term, living relations between persons and their place and participation in communities of practice. Thus identity, knowing and social membership entail one another." Lave & Wenger (1991), page 53

The importance of normative practice is also recognized by Rosenbrock (1981), who sees this as a constraint upon social change in technical design: while technical professionals adopt existing sociocultural value systems, human-centered methods and approaches to design can have little impact. But this perspective ignores the dynamic nature of sociocultural systems: social groups are not static, but are affected by their changing memberships.

In empirical studies, Curtis et al. (1988) stress the centrality of group learning processes and the critical role played by an ‘expert designer’ in communicating application domain knowledge to other team members. Expertise, according the Curtis et al. (1988) is not defined by technical knowledge, but by a valued operationalisation of technical knowledge in specific, local application domains. Communication and coordination activities are critical to design group functioning. In a study of small group design meetings, Olson et al. (1992) discovered that design teams spent only 40% of their time on direct discussions of design. The groups spent one-fifth of their time on "pure coordination activities" and *one-third* of their time on "clarification of ideas" - the sharing of expertise among group members. Walz et al. (1993) performed an observational study of design knowledge sharing and acquisition in videotaped design meetings; the authors "were surprised to see how important context-sensitive learning was to the design process". The issue was raised that much information was presented to the team during design meetings but never captured; Walz et al. (1993) do not examine whether this information was important and what prompted information capture.

Formal project documentation and design models - what Flor & Hutchins (1991) refer to as "external, structured representations" may play a significant role in the communication of situated knowledge in design teams. The way in which design information is represented fundamentally affects the way in which knowledge about that design is communicated and conceptualized (Simon, 1988; Winograd & Flores, 1986). Checkland & Scholes (1990) emphasize the centrality of the group learning which accrues from the production of joint models of the target object system during SSM workshops, to effective participation in organizational activity. Olson et al. (1993) observed that small design groups working with and without a group editor (which produced external structured representations of the design) generated more design ideas without the editor, but fewer and better ideas with it, indicating that the representations helped the supporting groups to remain more focussed on the core issues in the emerging design and to capture what was said as they proceeded. Flor & Hutchins (1991) note that external representations are critical in achieving intersubjective understanding; this concept is explored in the next section.

Intersubjectivity And Distributed Cognition

Curtis et al. (1988) conclude that "developing large software systems must be treated, at least in part, as a learning, communication and negotiation process." Designers have to integrate knowledge from several domains before they can function well. They identify the importance of designers with a high level of application domain knowledge: in their studies, these individuals were regarded by team members as "exceptional designers", who were adept at identifying unstated requirements, constraints, or exception conditions, possessed exceptional communication skills. Exceptional designers spent a great deal of their time communicating their vision of the system to other team members, and identified with the performance of their projects to the point where they suffered exceptional personal stress as a result. They dominated the team design process, often in the form of small coalitions, which "co-opted the design process". While these individuals were important for the *depth* of a design study, teams were important for exploring design decisions in *breadth* (ibid.).

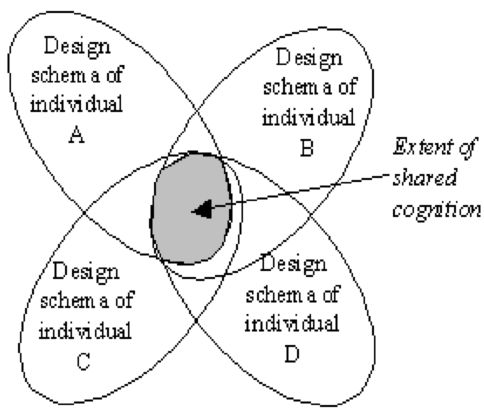

Figure 6. The Concept Of Shared Cognition (adapted from Laukkanen, 1994)

The acquisition of knowledge by design teams involves both *shared* cognition and *distributed* cognition. The concept of shared cognition is illustrated in Figure 6 and represents the extent of intersubjectivity (shared meanings) between organizational actors. Design depends upon intersubjectivity for effective communication between team members to take place (Flor and Hutchins, 1991; Hutchins, 1990, 1991, 1995; Orlikowski & Gash, 1994; Star, 1989). Technical system designers, "successful in sharing plans and goals, create an environment in which efficient communication can occur" (Flor and Hutchins, 1991). Orlikowski & Gash (1994), in a hermeneutic analysis of different interest groups' assumptions, knowledge and expectations of a new groupware technology, refer to intersubjectively-held mental models as "shared technological frames":

> " Because technologies are social artifacts, their material form and function will embody their sponsors' and developers' objectives, values, interest and knowledge regarding that technology" *(Orlikowski & Gash, 1994, page 179).*

The importance of intersubjectivity is emphasized by Lanzara (1983), who sees design as a dynamic process of framing and reframing of situations by a transactive process where different actors negotiate "their perspectives, values and (even!) facts". The different metaphors used (*c.f.* Morgan, 1986), at different times or by different actors, to frame the target system lead designers to emphasize differing *objects* through the design methods which they use. He gives the example of two extremes of metaphor in office systems design: the office may be seen as a "machine", in which case the object of design is <u>functions</u> and procedures, or it may be seen as a "community", in which case the object of design is conversations and <u>transactions</u> (Lanzara, 1983).

Little is known about how developers themselves perceive intersubjectivity or frames of reference (Flor and Hutchins, 1991; Orlikowski & Gash, 1994). Jeffries et al. (1981) stress the importance of ‘learning by doing’: experience enables concepts to be linked on the basis of the utility of keeping the concepts together and provides a coherent executive model of the processes of and of possible appropriate solution forms to design problems: in design teams this becomes a shared, intersubjectively-held design schema, which enables the team to function coherently.

The objects of different actors’ frames of reference can be seen as distinct and may be in conflict (Corbett, 1995). Orlikowski and Gash (1994) note that the frames of reference of IT managers and designers affect their decisions to adopt a particular information technology and to implement specific features of that technology; whilst those of business managers and users affect the way in which the technology is interpreted and used in business processes. A critical design activity is therefore that of making stakeholders’ frames of reference explicit and subject to debate, in the interests of intersubjectivity.

There is a trade-off between intersubjectivity and the exploration of design alternatives: if group members possess too much common ground, they may communicate more efficiently but there may be less of a tendency to explore alternative courses of action (Flor and Hutchins, 1991; Wilson and Canter, 1993). An experimental study by Rugs & Kaplan (1993) stressed the importance of goal congruence (i.e. intersubjectively-held goals) in group decision-making. As might be expected group (shared) goals facilitated greater *normative* influence upon decision-making (based upon social relations) and task goals facilitated greater *informational* influence (based upon evidence about reality). The implications for design teams are that effective consideration of organizational design requirements is only possible when there is a high degree of divergence between individuals’ models, whereas effective synthesis of solutions requires much higher levels of intersubjectivity. Viewed from this perspective, the separation of requirements analysis and design makes sense: in the early stages of design, members of a design-team are likely to hold diverging models of design requirements and goals and thus cannot effectively synthesize solutions; during later stages, levels of intersubjectivity may be higher and so solution synthesis could proceed more effectively. However, one should distinguish between co-operative design (where all design team members are working on a solution for the same subproblem) and coordinated design (where design team members synthesize solutions to different subproblems which are then brought together in a system solution): the former may require high levels of intersubjective understanding, the latter may function through distributed cognition.

Design activities do not just take place at the *individual* cognitive level, but also involve *distributed* cognition (Norman, 1991; Hutchins, 1990, 1991, 1995) in the social processes of negotiating design requirements and joint elicitation of design solutions. Distributed cognition implies interdependency between actors’ individual schemas:

> " Distributed cognition is the process whereby individuals who act autonomously within a decision domain make interpretations of their situation and exchange them with others with whom they have interdependencies so that each may act with an understanding of their own situation and that of others." *(Boland et al., 1994, page 457)*.

Hutchins (1990) describes the distributed mental models of the situation accessed by an airplane cockpit crew in plotting a course when control equipment broke down. No one actor held a complete model of the situation, but individual actors held both partial models of the solution and a process model which enabled them to coordinate other actors' partial models to reach a complete solution. Hutchins (1991) studied how the social organization of distributed cognition affects the cognitive properties of groups in a study of how communities arrived at shared versus differing understandings. He concluded that cognition in this type of situation is shared among agents in organizationally-prescribed roles and also among the artifacts that they use, such as work-procedures, charts, plans and routines for route-calculation - i.e. that models of how a situation may be handled are embodied in the artifacts used to expedite its handling. This echoes the work of the actor-network theorists (e.g. Callon, 1991; Latour, 1987; ,Law, 1992), in treating technological artifacts as 'non-human actors' in the analysis of the 'web' of distributed interactions in organizational decision-making.

Star (1989) addresses the process of combining evidence from different perspectives, in terms of how decision-group participants decide that sufficient, reliable, and fair amounts of evidence have been collected. Star discusses two studies, one of two different groups of physiologists and one of two different groups of biologists, making the following observations:

- Different groups can co-operate without having good models of each other. They can successfully work together while employing different units of analysis, methods of aggregating data, and different abstractions of data

- Different groups can co-operate although they have different goals, time horizons, and audiences to satisfy

- Coordination activities are supported by creating "boundary objects" which can be adapted locally to needs and constraints while maintaining a global identity.

The concept of boundary object is also raised by Norman (1992) who refers to cognitive aids used by commercial airplane crews. Individuals created artifacts to better understand the state of affairs in the cockpit, such as metal or plastic tabs that pilots move around the outside of the airspeed indicator to help remember critical settings. Crews used checklists to provide a mechanism for shared understanding and group memory. A member of the crew reads the list while others perform appropriate operations or checks - using this mechanism, the entire crew is informed about the state of the aircraft and a record of actions performed and the current state of the aircraft is kept. Designers in the study by Flor & Hutchins (1991) were observed to use external structured representations as a means of sharing knowledge.

Empirical evidence as to how design is managed in groups is inconclusive and scarce. There is evidence to show that intersubjective understandings are key to group design (Flor & Hutchins, 1991) yet there is evidence from studies of non-design group processes which involve coordinated action to indicate that *distributed* understanding - actors holding only partial mental models which are interdependent upon other actors' partial mental models for effective action - is also critical. There is some evidence to show that design is a collective *learning* experience: information systems designers jointly develop an intersubjective model of the system on which they base their design assumptions (Curtis et al., 1988; Flor and Hutchins, 1991; Reynolds & Wastell, 1996; Walz et al., 1993. But here is little evidence of how intersubjectivity and distributed cognition are achieved and maintained in design groups: what processes and criteria are involved in determining if information is significant and in sharing or emphasizing significant information? The concept of distributed cognition implies that design groups adopt satisficing, rather than optimizing behavior in design: individually, each team member does not have the cognitive capacity to understand the whole of a complex system, but individuals construct partial models of the problem situation, expecting that a co-operative design outcome will result from coherent group coordination in design.

So from an individual design focus of imposing structure upon a problem situation to facilitate convergence between a design problem-definition and potential partial solutions, we have reached a group design focus of coordinating and sharing partial design solutions. Lave (1991) suggests that the process of socially shared cognition should not be seen as ending in the internalization of knowledge by individuals, but as a process of becoming a member of a "community of sustained practice". Design groups need to maintain intersubjectively-held mental models of design goals and process, if they are to function effectively, yet each group-member may only hold a part of the knowledge and understanding necessary for design to take place. A group of individuals can pool their partial models to perform design activities through the mechanism of distributed cognition. This may be achieved through the shared meanings attached to artifacts used in common by a group - "boundary objects" (Star, 1989; Norman, 1992) - but there is only one study (Flor & Hutchins, 1995) which examines how such distributed models are maintained and this focussed on the extent to which external representations of the design were shared in a single experiment involving two programmers. Whilst some studies examine shared knowledge and learning in IS design (e.g. Curtis et al., 1988; Reynolds & Wastell, 1996; Walz et al., 1993), they do not examine how individuals frames of reference contribute to group perspectives of the design and how these group perspectives are constructed and maintained. The creation of 'communities of practice' (Lave & Wenger, 1991) is critical to coordinated design processes, but there are few studies of the mechanisms for achieving this in information system design. As software tools and methods are usually designed for individual problem-solving, they do not support those design processes which emerge from the social behavior of the development team or the organizational behavior of the company: we need research into the nature of group design processes to inform the provision of design tools.

Conclusions

This paper presented and compared two competing worldviews of design: design as organizational problem-solving and design as social cognition. Differences between the two worldviews are summarized in Table 4.

	Problem-solving	**Social Construction**
Individual	Problem definition and goal-determination (planning vs. emergent action)	Situated learning and constructing mental models of problems and solutions
Group	Deriving and validating consensus models	Learning and constructing joint models
Organization	Stakeholder negotiation of political design objectives, scope & strategy	Social construction of meaning and legitimacy; alignment of diverse interests
Perspective (re: the individual)	Exterior (reflected through objective models of a design, which exist separately from the designer)	Relationship between interior and exterior (reflected through external representations of mental models)
View of	Objective: Plans determine outcomes, to a	Constructionist: agents socially construct

Agency	greater or lesser extent.	their world, or Interactionist: Agent & their world co-constitute each other.
Representative Metaphors	"Surfacing" objectives; Design "rationale"; A "common vision"	"Framing" a design; "Constructing" a model; "External representations"
Management Focus	Achieving consensus and coordinating coherent action	Achieving intersubjectivity and collaboration in distributed cognition

Table 4: Comparison Of Two Worldviews Underlying Organizational IS Design

The "problem-solving" perspective takes as axiomatic that design is (more or less) objective: an individual explores a context (i.e. a problem space), to arrive at a design "solution" which is determined by that context. The individual "defines" a system solution to the perceived problem. The objective (external) nature of the problem context and the defined artifact or system of work activities is not challenged in any of the perspectives which are encompassed by this paradigm.

The social construction perspective is more radical in that it views design as the joint construction of *context* as well as *form* (of the information system or technological artifact). By engaging in normative sociocultural practices, designer construct a jointly-held view of the world (design context) which enables them to work cohesively in constructing an artifact or system which engages this worldview. The nature of this engagement has been touched upon in the work of Checkland (1981, Checkland & Scholes, 1990), who includes consideration of the Weltanschauung (worldview) of actors involved in a "problem situation" as part of his systemic approach to organizational problem-exploration, but who fails to engage with the social nature of this worldview.

It can be argued that both paradigms have a place in organizational information system design practice: they are complementary. On the one hand, designers must operate in a design-space which is ontologically viewed as having an existence independent of the design-team. On the other hand, designers must recognize that, through their group discussions and models, they construct a "problem context" which is very different from the context recognized by other actors in the organization. Any effective method of managing organizational IS design must encompass both of these views of action.

References

Ackoff, R.L. (1974) *Redesigning The Future*, Wiley

Alexander, C. (1964) *Notes On The Synthesis Of Form*, McGraw Hill, New York

Ball, L.J. & Ormerod, T.C. (1995) Structured and opportunistic processing in design: a critical discussion', Int. Journal of Human-Computer Interaction, Vol. 43, pp 131-151

Bijker, W.E., Hughes, T.P. and Pinch, T.J. (1987) (Eds.) *The Social Construction of Technological Systems*, *New Directions in the Sociology and History of Technology*, MIT Press, Cambridge, MA

Boland, R. and Day, W.F. (1989), The experience of systems design: a hermeneutic of organisational action, *Scandinavian Journal of Management*, 5,2 87-104

Boland, R J, Tenkasi, R V, Te'eni, D (1994) Designing Information Technology to Support Distributed Cognition, *Organization Science*, Vol 5, No 3, pp 456-475

Brown, J.S. & Duguid, P. (1992) 'Enacting Design for the Workplace' in P.S. Adler & T.A. Winograd (Eds.) *Usability: Turning Technologies Into Tools*, ACM Press, New York, NY

Burrell, G. (1983) 'Systems Thinking, Systems Practice: A Review', *Journal Of Applied Systems Analysis*, 10.

Callon, M. (1991) 'Techno-Economic Networks and Irreversibility', in J. Law (Ed.) *A Sociology of Monsters. Essays on Power, Technology and Domination*, Routledge, London, UK

Ciborra, C.U. (1987) 'Research Agenda For A Transaction Costs Approach To Information Systems', in R.J. Boland, and R.A. Hirschheim (eds.) *Critical Issues In Information Systems Research*, Wiley.

Clegg, C (1994) Psychology And Information Technology: The Study Of Cognition In Organisations, British Journal of Psychology, 85, pp 449-477

Corbett, J.M. (1995) 'IT And Work Design: The Negotiation Of Expertise' in Scarborough, H. (Ed) *The Management of Expertise*, London, MacMillan.

Corbett, J.M., Rasmussen, L.B. & Rauner, F. (1991) *Crossing the Border: The social and engineering design of computer integrated manufacturing systems*, Springer-Verlag, London

Curtis, B., Krasner, H. and Iscoe, N. (1988) 'A Field Study Of The Software Design Process For Large Systems', *Communications of the ACM*, Nov 1988, Vol 31, No 11, pp 1268-1287

De Marco, T. (1979) *Structured Analysis and System Specification*, Yourdon Inc.

Dorst, K. and Dijkhuis, J. (1996) 'Comparing paradigms for describing design activity', in N. Cross, H. Christiaans & K. Dorst (Eds.) *Analysing Design Activity*, John Wiley & Sons, Chichester, UK

Flood, R L (1995) 'What is happening when you problem-solve? A Critical Systems Perspective', Systems Practice, Vol. 8 No 2, pp 215-222

Flor, N.V. & Hutchins, E.L. (1991) 'Analyzing distributed cognition in software teams: a case study of team programming during perfective software maintenance', in J. Koenemann-Belliveau, T. Moher & S. Robertson (Eds.) *Empirical Studies of Programmers - Fourth Workshop*, Norwood, NJ: Ablex

Galliers, R.D. (1987) An Approach To Information Needs Analysis, in Galliers, R.D. [Ed.], Information Analysis, Selected Readings, Addison-Wesley

Galliers, R.D. (1993a) SSM??

Gane, C. & Sarsen, T (1979) *Structured Systems Analysis: Tools and Techniques*, Prentice-Hall, New Jersey

Gasser, L. (1986) The integration of computing and routine work, *ACM Transactions On Office Information Systems*, Vol. 4, pp 205-252

Gilhooley, K.J. (1989) 'Human and Machine Problem-Solving, A comparative Overview' in K.J. Gilhooley (Ed.) *Human and Machine Problem-Solving*, Plenum Press, New York

Guindon, R. (1990a) 'Designing the design process: Exploiting opportunistic thoughts', *Human-Computer Interaction*, No. 5

Guindon, R. (1990b) 'Knowledge Exploited By Experts During Software System Design', *International Journal of Man-Machine Studies*, Vol. 33, October 1990, pp 279-304

Hutchins, E. (1990), 'The Technology of Team Navigation', in J. Galagher, Robert E. Kraut, and Carmen Egido (eds.) *Intellectual Teamwork: Social and Technological Foundations of Cooperative Work*, Hillsdale, NJ: Lawrence Erlbaum Associates, Publishers. pp. 191-220.

Hutchins, E. (1991), 'The Social Organization of Distributed Cognition', in Lauren B. Resnick, John M. Levine, and Stephanie D. Teasley (eds.) *Perspectives on Socially Shared Cognition*, Washington, DC: American Psychological Association. pp. 283-307.

Hutchins, E. (1995) *Cognition in the Wild*, MIT Press, Bradford

Jeffries, R., Turner, A.A., Polson, P.G. & Atwood, M.E. (1981) 'The Processes Involved In Designing Software', in J.R. Anderson (ed.) *Cognitive Skills And Their Acquisition*, Lawrence Erlbaum Associates, Hillsdale, New Jersey

Khushalani, A., Smith, R. & Howard, S. (1994) 'What Happens When Designers Don't Play By The Rules: Towards a Model of Opportunistic Behaviour In Design', Australian Journal of Information Systems, May 1994, pp 13-31

Lanzara, G.F. (1983) 'The Design Process: Frames, Metaphors And Games', *in* U. Briefs, C. Ciborra, L. Schneider (eds.) *Systems Design For, With and By The Users*, North-Holland Publishing Company

Latour, B. (1987) *Science in Action*, Harvard University Press, Cambridge, MA

Lave, J. (1991) 'Situating Learning In Communities of Practice' in L.B. Resnick, J.M. Levine, S.D. Teasley (Eds.) *Perspectives on Socially Shared Cognition*, American Psychological Association, Washington DC, pp 63-82

Lave, J. & Wenger, E. (1991) *Situated Learning: Legitimate Peripheral Participation*, Cambridge University Press, Cambridge, UK

Law, J. (1992) ‘Notes On The Theory of the Actor-Network: Ordering, strategy and Heterogeneity’ *Systems Practice*, Vol. 5, No. 4, pp 379-393

Law, J. & Callon, M. (1992) ‘The Life and Death of an Aircraft: A Network Analysis of Technical Change’ in W.E. Bijker and J. Law (Eds.) *Shaping Technology/Building Society: Studies In Sociotechnical Change*, MIT Press, Cambridge, MA

Lawson, B. (1990) *How Designers Think*, The Architectural Press, London, UK (Second Edition).

Mackenzie, D.A. & Wajcman, J. (1985) ‘Introduction’ to Mackenzie, D.A. & Wajcman, J. (eds) *The Social Shaping Of Technology*, Open University Press, Milton Keynes

Malhotra, A., Thomas, J., Carroll, J. & Miller, L. (1980) ‘Cognitive Processes In Design’, *International Journal of Man-Machine Studies*, 12, pp 119-140

Matthiassen, L. & Stage, J. (1992) ‘The Principle of Limited Reductionism In Software Design’, *Information Technology & People*, Vol. 6, No. 2, pp 171-185

Mayer, R.E. (1989) ‘Human Nonadversary Problem-Solving’ in K.J. Gilhooley (Ed.) *Human and Machine Problem-Solving*, Plenum Press, New York

Methlie, L.B. (1980) ‘Systems Requirements Analysis - Methods And Models’, in H.C. Lucas, F.F. Land, J.J. Lincoln, K. Supper (Eds.) *The Information Systems Environment*, Proceedings of IFIP WG8.2, Bonn 1979, North-Holland Publishing Company

Mintzberg, H. & Waters, J.H. (1985) ‘Of Strategies, Deliberate and Emergent’, Strategic Management Journal, vol. 6, pp 257-72)

Morgan, G. (1986) *Images Of Organization*, Sage Publications, California

Newell, A. & Simon, H.A. (1972) *Human Problem-Solving*, Prentice Hall

Norman, D.A. (1986) ‘Cognitive Engineering’, *in* D.A. Norman & S.W Draper (Eds.) *User-Centred System Design: New Perspectives on Human-Computer Interaction*, Erlbaum Associates, Hillsdale, New Jersey

Norman, D.A. (1988) *The Psychology of Everyday Things*, Basic Books, Harper-Collins, USA (Previously published as: The Design of Everyday Things. 1990. New York: Doubleday)

Norman, D.A. (1991) ‘Cognitive Artifacts’, in J.M. Carroll (Ed.) *Designing Interaction: Psychology At The Human-Computer Interface*, Cambridge University Press, UK, pp 17-38

Norman, D. A. (1992) ‘Turn Signals are the Facial Expressions of Automobiles’, Addison-Wesley Publishing Company, Inc.. Reading, MA

Norman, D. A. (1993) ‘Cognition in the Head and in the World: Introduction to the Special Issue on Situated Action’, *Cognitive Science*, Vol. 17, No. 1, pp 1-6

Olson, G.M., Olson, J.S., Carter, M.R., Storrosten, M. (1992) ‘Small group design meetings: an analysis of collaboration’, *Human-Computer Interaction*, Vol. 7, pp 347-374

Orlikowski, W.J. & Gash, D.C. (1994) ‘Technological Frames: Making Sense of Information Technology in Organizations’, ACM Transactions on Information Systems, April 1994.

Preston, A (1991)??

Reynolds, M. & Wastell, D. (1996) ‘Breaking Down The Social Defences: Enabling Learning With Transitional Objects’ in *Proceedings of 4th European Conference on Information Systems*, Lisbon, Portugal, July 2-4 1996

Rosenbrock, H.H. (1981) ‘Engineers And The Work That People Do’, *IEEE Control Systems Magazine*, Vol 1, No.3 pp 4-8 September 1981. *Reprinted* in Paton et. al. (1984) *Organisations: Cases, Issues and Concepts*, Paul Chapman Publishing, London

Rugs, D. & Kaplan, M.F. (1993) ‘Effectiveness of informational and normative influences in group decision making depends on the group interactive goal’, *British Journal of Social Psychology*, Vol. 32, pp 147-158

Simon, H.A. (1960) *The New Science of Management Decision*, Harper & Row, New York

Simon, H.A. (1973) ‘The Structure of Ill-Structured Problems’, *Artificial Intelligence*, No. 4, pp 145-180

Simon, H.A. (1981) *Sciences of The Artificial*, Second Edition, MIT Press, Cambridge, MA

Simon, H.A. (1988) ‘Scientific Discovery As Problem Solving’, Peano Lecture, reprinted in Egidi, M & Marris, R. (Eds.) *Economics, Bounded Rationality and the Cognitive Revolution*, Edward Elgar Publishing Limited, Aldershot, UK (1992)

Star, S. L. (1989), ‘The Structure of Ill-Structured Solutions: Boundary Objects and Heterogeneous Distributed Problem Solving’, in L. Gasser and M. N. Huhns (eds.) *Distributed Artificial Intelligence*, Vol. II. San Mateo, CA: Morgan Kaufmann Publishers Inc., pp. 37-54.

Star, S. L. (1992), ‘The Trojan Door: Organizations, Work and the “Open Black Box” ‘, *Systems Practice*, Vol. 5, No. 4.

Suchman, L. (1987) *Plans And Situated Action*, Cambridge University Press, MA, USA

Taylor, F.W. (1947) *Scientific Management*, Harper & Row

Truex, D. and Klein, H.K. (1991) ‘A Rejection of Structure as a Basis for Information Systems Development’, in R.K. Stamper, P. Kerola, R. Lee and K. Lyytinen (eds.) *Collaborative Work, Social Communications and Information Systems*, Elsevier Science Publishers B.V. (North Holland) IFIP 1991

Turner, J.A. (1987) ‘Understanding The Elements Of Systems Design’, in Boland, R.J. and Hirschheim, R.A. (eds.) *Critical Issues In Information Systems Research*, Wiley.

Veryard (1986)??

Walz, D., Elam, J. & Curtis, B. (1993) ‘Inside A Design Team: Knowledge Acquisition, Sharing and Integration’, *Communications of the ACM*, October 1993, Vol. 36, no. 10, pp 63 - 77

Winter, M.C., Brown, D.H. & Checkland, P.B. (1995) ‘A Role For Soft Systems in Information Systems Development, *European Journal of Information Systems*, 4 pp 130-142

Wood-Harper, A.T. (1990) ‘Characteristics of Information Systems Definition Approaches’, *in* R.L. Flood, M.C. Jackson, & P. Keys (Eds.) *Systems Prospects: The Next Ten Years of Systems Research*, Plenum Press, New York

Yourdon, E. (1989) *Modern Structured Analysis*, Yourdon Press, Englewood Cliffs, New Jersey

Yourdon, E. (1993) *Yourdon Systems Method: Model-Driven Systems Development*, Yourdon Press, Englewood Cliffs, New Jersey

Yourdon, E., and Constantine, L.L. (1975) *Structured Design*, Yourdon Press, Englewood Cliffs, New Jersey

Last updated 12/01/2007 17:06:51