A Bottom-Up Approach to Design Coordination

Michael Cumming
Delft University of Technology
Faculty of Architecture
Berlageweg 1, 2628 CR Delft
The Netherlands

ABSTRACT

The design industry like any other industry could benefit from process support for its practitioners. However, what constitutes legitimate process support for designers is a difficult research question. In order to support designers in their processes, first one needs to know what designers do. This is not only a question of knowing what designers do in general, but also what particular designers do in their own individual design practice. How to find out what designers do, that is, how to acquire knowledge about their design processes, is also a difficult research question. Providing process support for designers is seen as a task, which cannot be divorced from design process acquisition. That is, one is unlikely to provide process support, without acquiring an understanding, what individual designers actually do, and conversely, one is unlikely to acquire an understanding of what designers do, without providing process support for designers. An application whose goal it is to provide both design process support for designers working in collaborative design teams, and also acquires a particular type of design process representation of design processes, is described.

1 INTRODUCTION

The design industry is similar to many others, in that there is an increasing concern for the quality of processes. Processes are seen to structure, and inform all industries. In fact, many business theorists propose that processes are really, what define an enterprise (Evan, 1993). With such process-centric definitions of business enterprise, quality of process tends to be seen as a dominant factor in determining the quality of the whole enterprise.

This concern for process quality is not only motivated by concerns for increased product quality, which is an important consideration in design, but is also related to other factors such as worker productivity, stakeholder involvement, process improvement, and business re-engineering.

In many industries, the idea of process control is central to attempts to improve quality. That is, processes should conform to some conception of a preferred process. The idea is that in order to promote quality, preferred processes should be conceived. This is the idea, for instance, in the software engineering process model called the Capability Maturity Model (CMM). The CMM has become an influential standard for assessing and improving software processes, used for modeling, defining, and measuring the quality of the processes used by software professionals. The process in improving how software design processes are managed, are seen to go through five stages: initial, repeatable, defined, managed, and optimizing (Software Engineering Institute, 1995). Designers are trained to put much of their energy towards the goal of creating interesting, high quality products. However, what constitutes quality of a design process is much less clear.

Quality of design, especially in the long run, depends both on excellences of product, and of process. Such an attitude has relevance both from the designer's, and the manager's point of view. Quality of process and product are not independent from each other: good processes tend to lead to good products, and vice versa. For instance, if employees enjoy their work and are treated well, they tend to do good work - that is, produce good products. On the other if people get to work on good projects that involve the design of interesting products, designers tend to enjoy themselves more, and therefore enjoy their process more. Good products can exist without good processes, and vice versa, but generally, the two tend to be mutually reinforcing.

The design process, unlike many processes in other industries, is however seen by designers, and most experts in specialized knowledge domains, as something, which they should have some autonomy to control themselves. This is especially true for creative design processes. In order to control some process, it is presumed that it is possible to predict the nature of such processes, before they occur. Creative processes, by definition, tend to involve a high degree of unpredictably, and lack of anticipation. Therefore, the idea that process control will lead towards quality does not appear to likely to be a popular idea among creative designers, since it suggests a type of design that is predictable, and has well documented processes. This tends to be the case in routine, rather than creative design processes. Creative design usually exhibits a very dynamic nature, with a large degree of opportunistic behaviors on the part of designers. Surely, attempts at improving design process quality should not come at the expense of design creativity.

Design processes, especially collaborative design processes, are not only private cognitive acts that take place in the minds of individual designers, but also the social processes that occur between designers, and other stakeholders in a design process. In the current research, such social processes are assumed to be of importance and significance, in shaping the results of collaborative design. They also tend to be much easier to document and coordinate, since they involve behaviors, such as communications, that have a tangible, easily documented reality. It is not the intention here to take a behaviorist position, in which only externally observable behaviors are given scientific credibility. Obviously, designers do engage in cognitive, mental acts that do affect collaborative design processes. It is possible to support such cognitive processes in credible ways.

The position here is that design also involves social behaviors. When the emphasis becomes on social behaviors, the idea of design coordination - how to coordinate your work with the work of others becomes important. Providing mechanisms that support such coordination is the design support aspect of this research.

Given that process control may affect design creativity, is there any conceivable way of providing process support for designers, without involving process control, or risk reducing design creativity? One answer may lie with concern for overall process coordination, rather than control of specific, predefined processes. Here the idea is allow designers the freedom to pursue any type of process they wish, but support them in coordinating these processes with the processes of others.

Process support usually involves constructing some idea, or model about what sort of processes that designers actually do. There are two ways of doing this. One is to observe the processes of specific designers, or design teams, and attempt to generalize the results, such that they could be seen to legitimately inform a wider range of design processes. Another is not to attempt to generalize, but to attempt to acquire specific detailed knowledge about how individual designer actually do. Finding out what designers actually 'do' is also a difficult research question. On a purely practical level, it is not easy to gather such information, regardless of the knowledge acquisition method one might use.

Another problem is the general problem of research bias. What a researcher ends up finding is often dependent on what sort of things he expects to find. Such a bias can often be beyond the conscious awareness of the researcher. Such a bias is often not caught by her research peers, because they may too be influenced, from the same types of preconceptions.

Another is that what designers do, may be a result of both intentional factors, or the result of emergent factors. The intentional factors are ones that the designer may be consciously aware of doing. The emergent ones that may have arisen with no ones conscious intention, but may have caused by complex interactions between various independent agents. In these cases, the particular 'social field' or 'culture of influence' in which the design team may be embedded, may have had as much influence, as the influence of individuals.

This research takes, as one of its primary ideas is that providing process support for individual designers, and gathering reliable information about what these individual designers actually do while performing design, are mutually dependent research tasks. That is, it is unlikely that one can provide adequate process support for individuals, without an intimate knowledge of what sorts of processes the individual already engages in. This argument also works the other way around: it is unlikely that one can acquire process knowledge that has some empirical basis, relevant for an individual, unless one provides real process support while doing this. The reason being is that collaborative is already an activity that is taxing on many levels for designers: cognitive, social, and technical. In order to gather process information, by a researcher, usually means that the designers must disrupt her normal processes. This intrusion tends to increase the burden of collaborative design.

This research concerns the design of an application that attempts to do both of above things. That is, it attempts to provide design support for individual designers. It does this through enabling a simple peer-to-peer coordination process between design collaborators. It also enables the acquisition of a particular kind of process model, as a by-product of this coordination process.

2 MOTIVATION

Motivation for this work derives from some interesting situations that the author observed many times, in architectural practice.

One is how some design processes can take very curious paths, without anyone in particular appearing to be responsible for their direction. This is not to say that such processes are necessarily chaotic or destructive, just that the parties which traditionally are thought to control design processes, such as lead designers or clients, may not be in a position to control the direction that processes eventually take. This is curious situation because design, almost by definition, is usually thought of as involving intentional acts by individuals, or groups of individuals. The idea that design processes can sometimes just ‘happen’ is unusual. Such situations could be called, from a complex systems perspective, instances of unintentional process emergence.

Another, interesting situation which the author experienced, is how design processes and products can sometimes lead ‘separate lives’, such that the nature of one does not necessarily reflect on the nature of the other. For instance, sometimes the design product resulting from a collaborative design process can seem to be of high quality. Normally this would suggest that the design process, during which it was created, would also be of high quality. However, this is not always the case.

The reverse sometimes happened as well: the design process would be fine, while the resulting product is judged not to be very successful. This is a curious situation because often process and product are presented as being mutually dependent, in which the product structures the process, and vice versa.

This research is motivated by the goal of explaining both of these situations. How design processes can emerge in practice without necessarily someone’s intention, and what the connection is between designers’ processes, and the products they design.

3 BACKGROUND

3.1 **Collaborative design processes, and their support**

3.1.1. Nature of collaborative design

Collaborative design is a common way of designing, yet it tends to be a very complex activity. Collaborative design depends on the successful interaction of many different parties. The nature and outcome of these interactions can be quite *ad hoc*, and specific in nature, and therefore difficult to predict, and to generalize. Design problems are also becoming more complex, with increasing integration demanded between diverse and possibly novel functional requirements. Increased complexity in design has both social and technical aspects. Not only are the technical problems becoming more difficult, such as learning to work with new materials, or learning to cope with changing regulatory environments, but the social demands that they bring is also changing. People from different cultures, who may have never worked together before, are brought together and expected to be quickly bridge striking cultural differences and become productive with one another.

The concept of the 'stakeholder' is becoming more prominent in collaborative design. Here people, who previously may have had little input into a design process, are demanding that their concerns and opinions to be heard, and that these concerns be somehow incorporated into design products (Evan, 1993).

With design, processes where the input of stakeholders is taken seriously, the situation can arise in which any stakeholder within a design process could conceivably affect that process. Therefore, the study of design processes should include all parties who are stakeholders in a collaborative design process.

Stakeholders come to design process with various levels of design experience and expertise can have profoundly diverse conceptual perspectives on the design process and product. This variation is not only a function of their roles within the design process, their professional and educational experience, but is also function of their own personal histories. Since these influences vary so much, and can come vary in so many unforeseen dimensions, the only way of understanding the motivations of stakeholders, and the effect they might have on a design process, is to communicate directly with them.

It seems clear that collaborative design is fundamentally a collaborative process, whose outcome depends on the interactions of a large number of parties. If a new person fills this role, and assuming that the previous person informed the new person how to carry on this practice, then it likely

1.1.1 Design methodology

There have been many models of the design process that have arisen from design domains such as engineering, architecture, and industrial design. According to Roozenburg and Cross (1991), in engineering design these models have converged on what they call a 'consensus' model, based on German engineering theory (Roozenburg and Cross, 1991). Such a consensus model involves a rational, linear, progressive series of tasks in which activities are grouped into four phases: clarification of the task, conceptual design, embodiment design, and detail design.

In architectural design circles, overly prescriptive linear process structures were replaced by spiral models, in which design could revisit tasks and iterate processes (Cross, 1993).

Models by Darke questioned the idea that exhaustive problem analysis and specification is always a necessary precursor to design synthesis (Darke, 1984). Here the unpredictable ideas and biases that individuals might bring to a design process were seen to play an important role.

Greater flexibility in the order of design activities was promoted by Guidon. He suggested that so-called opportunistic behaviors, in which design activities were interleaved in complex behavioral and cognitive structures, are common in design. These were seen not as corruptions of a rational process, but rather as appropriate designer responses to the ill-defined nature of design, in the context of limited cognitive resources (Guindon, 1990).

The work of Donald Schön criticized the notion that design is a process that is ruled by explicit and rational problem-solving knowledge, and argues that designers usually know more than they can say (Schön, 1983). Schön calls this a kind of 'knowing-in-practice', which is a form of tacit knowledge. Instead of a deterministic process that is driven by technical requirements, he views design more as iterative meaning-forming one. In such a model of design, designers alternate between action, and reflecting on that their action, in order to construct for themselves, meaningful representations of design problems and solutions.

The work of Louis Bucciarelli also deviates substantially from standard accounts of collaborative design (Bucciarelli, 1994). He argues that iterated social processes, such as narrative construction, are seen in collaborative design situations. In constructing stories, design teams attempt to make sense of their design problems, and to imagine plausible solutions. He also argues against viewing the structure of objects, such as those of the artifacts that are being designed, as appropriate in structuring a collaborative design process. Instead, he focuses on the social processes themselves, as being the most relevant factor in determining how design processes actually turn out. Design processes can be characterized by their level of innovation. The standard classification scheme involves the categories creative, innovative, and routine. See for instance (Dym and Levitt, 1991). These levels either can be decided at the outset, or can be an emergent product of the design processes themselves. These categories are not fixed - there exists a continuum between design processes that might change substantially from design project to project, to those, which are quite stable and exhibit little change. Providing design process support in situations where the processes never change is much easier than in situations where they do.

Innovation and creativity imply unpredictably in a design process. That is, the greater the design team desires to pursue innovative design processes, the greater is the uncertainty among the design participants, about how a design project should proceed and how it might turn out. Therefore, the most challenging end of design process support is where the participants are highly motivated to pursue creative design processes.

Design situations sometimes present designers with opportunities, both cognitive and social that designers should be prepared to recognize, and exploit opportunistically. Yet, it is possible to conceive of designing a design process. From a design management point of view, how a design process is conceived and how it progresses is an important consideration.

However designing a design process is a difficult task. For one thing, it seems to require a model of the design process that a wide community of designers would be willing to submit to, and be able to follow instinctively. No such model yet exists in the field of design methodology. This also tends to assume that the design process is something, which it is possible to design and then to be followed, much as a building product representation is a model, which is capable of being constructed.

Designing a design process also includes the problem of authorship. Is this a model that is designed by one party and imposed on another, or is it a model, similar to that of a designed product model, which requires the input of many collaborators, to make it work. If it requires the input of many parties, which seems sensible, then one runs into the recursive problem of designing an unpredictable process, by another unpredictable process.

3.1.2. Social processes in design

Usually design process models assume than design is a rational, problem-solving process, in which the cognitive abilities of individual designers, are paramount in coming to appropriate solutions. Less attention is paid to the complexities of collaborative design, in which a large number of people, with possibly divergent conceptual outlooks on the process and product, must learn to interact productively.

According to Whitney, it is possible to view design process in two, quite different ways. One, as a technical process to be accomplished, and two, as an organizational process to be managed. The first is tends to focus on the individual whereas the second on focus on the group (Whitney, 1990).

In collaborative design, both these aspects are important, and are worthy of support. It is perhaps in the complex interactions between the technical, and the social or organizational aspects of a design process, which presents the biggest challenges to a fully integrated design methodology.

Collaborative design involves many parties acting out a variety of roles. These roles may be formally assigned, and with clear-cut responsibilities and processes, or they may informally adopted by the participants themselves while the process proceeds. Such informal role-adoption was noted by Cross and Cross in group design protocols (Cross and Cross, 1996).

It can also be seen in social groups in general, that dominant and submissive social roles, such as leadership positions, may not be a simple matter who gets assigns to do what. If often involves an emergent social process of role-adoption. How such role emergence works is a complex process involving necessity: what roles ought to filled, opportunity: what roles are open for filling, and competition: who else might be suitable candidates to occupy the same role.

3.1.3. Design meetings

One of the most important settings of collaborative design is the design meeting. Here participants in the collaborative design process not only get together to discuss design progress which may be accomplished elsewhere, but they also design while in the meeting. Design meetings exhibit many characteristics that make it a suitable social and technical context for discussing, constructing, and resolving design problems and solutions. Some basic characteristics of design meetings are:

- A variety of different stakeholders and perspectives may be present. Some or all of these attendees may participate, and contribute to the meeting.

- A wide variety of social and technical processes may be seen within the meeting: discussions, presentations, problem solving, process management, brainstorming, critiques, etc.
- The meeting may or may not have an agenda, and this agenda may be adequately addressed during the meeting.
- The meeting may take a short time or a long time to complete. The length of time the meeting takes is not necessarily an accurate indication of the productivity of the meeting.
- Within the meeting, there may be pronounced hierarchical power relationships between attendees. These relationships may or may not be visible within the meeting, and they may or may not be helpful in coming to appropriate design solutions.
- A wide variety of products may result from the meeting. These products may or may not be accurately recorded and saved for later use. Such products could include decisions on processes and products, design directions, general strategies, relationships between people, information informing future relationships, business contacts, etc.

In order to provide design support tools involving social processes in design, there must be some shared understanding between designers who might use the tool, that the tool will help them perform design in a manner to which they find mutually acceptable. In order to come to such a consensus, there must be actual empirical data as to what designers actually do when they perform design. Such data is not only difficult to obtain, but it also tends to be biased by the theoretical expectations of the researchers who analyze it.

In order to expect real design support, designers should expect that the design support tool have some close relationship to the particular way of working. The way of working may be specific to the designer using the tool. It seems that designing tools to suit certain roles, rather than certain individuals, is likely to be seen as too impersonal and coarse a distinction, by its users.

This suggests that what is most important in process support is not so much a representational problem, but rather a knowledge acquisition problem. The design support must be able to learn something about how the designer works, rather than impose a design process theory on designer, which might not be particularly relevant or useful.

One of the most useful techniques that design theorists has used to gather empirical data about design behaviors and cognition, is the protocol study. See, for instance (Ericsson and Simon, 1980), (Akin and Lin, 1995), and (Cross, Christiaans, and Dorst, 1996). Protocol studies are seen as being extremely useful in design research contexts. However, they do have some disadvantages. First, they tend to view design processes as private cognitive acts that necessarily involve technical problem-solving tasks. Secondly, the protocol studies themselves are extremely time consuming to perform. The ratio between the time to complete the protocol, and the real-time duration of the design process it analyses, usually works out to about ten-to-one. Finally, they usually involve a single high-level semantic interpretation of what occurs during a design process. Since collaborative design involves multiple perspectives, all of which may be simultaneously valid, single interpretations, however well reasoned and well supported, remain as singular interpretations.

For these reasons, protocol studies are seen in this research as less useful in the real-time complexities of ordinary collaborative design practice. Instead, the focus will be the communicative, message-passing activities that designers perform during design. Instead of a cognitive approach, this could be called an 'interaction-based' approach.

An important idea in this research is that you cannot provide informed and relevant process support for particular designers, unless one acquires an understanding of the actual processes these particular individuals normally perform in design practice. Such an approach could be termed: 'local support requires local knowledge'.

One, of course, could try to generalize process information gathered from individual designers, such that they could be seen as relevant for other types of designers, design contexts, and design problems. This presumably is the intent of scientific study in general, to attempt to make supportable generalizations that apply to groups of instances, rather than merely to single ones.

There is also a basic problem with the recursive nature of design support. In order to provide any kind of support, one must have some knowledge, or some theory, about the design process. Such a theory may, or may not, turn out to be a supportable generalization of design practice. Indeed, design researchers are unlikely to be fully aware of all the theories, or assumptions, that informs their research.

There does not appear to a 'consensus' design process model, that a variety of designers, design managers, design methodologists could all subscribe. Such a consensus could inform all those involved in the design industry what are the essential features of design processes, and whether these features are generalizable, and relevant to all manners of designing, in all design situations.

With the diversity of stakeholders that might be found within a design process. This diversity is both a handicap to team performance that must be overcome, but it is also an important opportunity as source of diverse ideas. Diverse ideas tend to be a requirement in providing well-balanced solutions to complex multi-dimensional problems. Given this variety of perspectives, it seems implausible that any one methodology, even if it were possible for a design team to follow, and even if it were appropriate for the situation, that a design team would consent to use such a methodology. Of course, this situation could change with the invention of better design methodologies.

Collaborative design support should not involve prescriptions that tend to define – implicitly or explicitly – existing design methodologies, but should be supportive in other ways. Helping designers coordinate their action, rather than prescribe it, is the general strategy to be followed.

3.2 **Coordination theory**

Coordination is a new discipline that has been developed to help explain and manage complex new collaborative situations, which tend to overwhelm existing process management theory and technique. See for instance (Klein, 1998), and (Malone and Crowston, 1992) for excellent overviews.

Collaborative design is a prime example of a situation in which the ideas of coordination theory are relevant. Klein gives the example of the design and manufacturing of a commercial jet. Here the design process must necessarily involve the participation of many thousands of people and organizations, who are distributed geographically, as well as according to task and perspective (Klein, 1998).

With the design and manufacture of such a complex artifact as a commercial jet, the expectations that the final product will successfully meet a large number of possibly conflicting design requirements, are great. The process interdependencies involved are so great that they tend to overwhelm traditional design and management techniques. Coordination of action is required, according to Klein, when distributed activities, such as found in collaborative design, are interdependent.

Coordination science aims to be a general science. It does not just deal with specific processes, relevant for only certain industries. It attempts to create a general theory that is applicable to a wide variety of problems that require process coordination. The reason it was developed in the first place was to be address problems of such complexity, and of such importance, that existing sciences and techniques were not adequate.

The most suitable definition of coordination is that provided in (Malone and Crowston, 1992) 'the act of working together harmoniously'. Malone and Crowston also provide a list of technical definitions others have proposed for the term.

According to Klein (1998), support for coordination can be divided into three layers, built on top of each other:

Collaboration: allowing participants to collaboratively update some shared set of decisions (this involves support for tele-conferencing, etc.)

Communication (the lowest layer): allowing participants in the decision process to share information (this involves networking infrastructures).

All three of the above layers are relevant for design. Designers in collaborative design situations must obviously communicate, in order to perform their jobs. They do this using whatever synchronous and asynchronous tools they have available. These may range from face-to-face conversation such as what happens in design meetings, to e-mail, to groupware tools.

It is also clear that designers must collaboratively update a set of shared decision. This would involve for instance, integrated product models in design. The third (top-most) layer is the end goal of the collaborative exercise. Here what the desired effect is not known before the actual design process takes place. In creative design situations, the shape and nature of the design result may not be known until well into the design process.

According to Klein, processes can be represented using two major classes of process representations:
IPO (input-process-output): These representations describe process as series of tasks that take several inputs (representing both control and data) and produce one or more outputs that are then routed to other tasks

Speech-act based: these representations take the alternative approach of modeling cooperative work as being built up of prototypical loops of requesting, accepting, completing and accepting delivery of tasks. Each loop represents a task, with the customer on the left, and the performer on the right. See also (Winograd and Flores, 1987) for more on this commitment-based coordination representation, and also (Medina-Mora et al., 1992) for a commercial application called the 'Coordinator' which uses the same approach, and applies it to general business processes).
This distinction is important to note because the first type of process representation is much more common than the second type, which remains relatively obscure.

3.2.1. Coordination and language

From one social psychologist's point of view, coordination of action is also the primary purpose of language. According to Clark (1996): Joint actions are created when people coordinate with each other. Why should they coordinate? The reason according to Thomas Schelling (1960) is to *solve coordination problems*. Two people have a coordination problem whenever they have common interests, or goals, and each person's actions depend on the actions of the other. Some of the issues that coordination theory address is: How can overall goals be subdivided into actions? How can actions be assigned to groups or to individual actors? How can resources be allocated among different actors. How can information be shared among different actions to help achieve the overall goals. Obviously, such issues are relevant in collaborative design, and are important in the design of computer-based collaboration tools for domains other than design. The type of approach towards collaborative design process support taken here is one that focuses on the coordination of action, rather than the action itself.

The use of language has similarities with that of collaborative design, in that both involve highly interactive processes. Design participants must interactively deal with the demands that their collaborators place upon them, much as a person having a normal conversation must attend to the demands of their conversational partners. In such interactive processes, the participants may not have a clear idea of where a verbal interchange, such as an ordinary conversation, is headed. Such foresight would require prior knowledge of the motivations and goals, of the party they are conversing with. Such knowledge instead, tends to be gathered incrementally by each party, within the process itself - that is, while the people hold their conversation.

A similar situation seems to occur in collaborative design situations. People do come to collaborative design processes with expectations how other will likely behave. However, these expectations still need to confirm with each design collaborators, while in the process of interacting with them.

3.2.2. Coordination and commitment

The work of Terry Winograd incorporates well-known criticisms of the rationalist approach to design, cognition, and intelligence (Winograd and Flores, 1987). He emphasizes, like Schön and Bucciarelli, the social processes that are required for social groups to come to some common understanding. He sees the interactive nature of language use, such as described in speech act theory (Searle, 1991), to be a primary factor in such social processes.

His work has much to say about coordination in design processes. He views all interactive communication as involving commitments to complete future action. Once commitments have been communicated, these exists a social commitment to complete them, or face the social costs of non-completion. He sees this as a basic characteristic in all use of language, in which usage of words involves commitment to a social constructed sense of what the words 'mean'. This idea can be applied to both human interactive systems, such as collaborative design teams, as well as computer systems, in which meaning of words and communications is a highly interactive process that demands continuous negotiation.

From a coordination science perspective, no matter what the content that a design process might include, designers need to coordinate their work. This involves many social and cognitive processes, the most basic of which seems to be communication. This involves design participants, giving each other adequate information with respect to their activities, such that they have an adequate understanding of what each other is doing, and what each other is planning on doing. Such a process is highly interactive.

Design coordination implies communication between design stakeholders. This communication can take many forms, such as normal interpersonal interaction, or computer-mediated forms, and normally it involves use of language, preferably in face-to-face settings. It seems plausible that the more channels of communication that are open to designers, the greater the chance is that design coordination will occur.

Design coordination can be achieved either from top-down or bottom-up processes. Top-down processes involve high-level parties, such as person with some power and influence, defining a set of intentions, and imposing them on others, normally lower down in a social or technical hierarchy.

Top-down coordination is not necessarily a bad thing. People, as they gain influence within a design process often have as their legitimate responsibility, the task of guiding and steering the design process. Often this control is actually welcomed by those lower in the hierarchy, since it can provide structure and purpose to a complex process, and can make their jobs much easier.

Another form of top-down coordination, is when groups of individuals, attempt to achieve a consensus. In forming a consensus, many people may be required to give their explicit or implicit approval. This consensus, if it is actually honored by its participants, in effect begins to resemble the top-down intentions of a single person. Forming a group consensus can be a powerful way of influencing the behavior of other in the future, and thereby reducing the unpredictably of design.

With bottom-up coordination, people interact in such a way that their local processes become coordinated. If enough people behave in this fashion, then higher-level entities, such as design teams or design projects, also become coordinated. Such processes do not necessarily require a top-level entity, although top-level influence can sometimes be useful in encouraging lower level cooperative behaviors. The results derived from bottom-up coordination are not necessarily any better than those of top-down approaches. Sometimes bad can emerge from bottom-up processes, as well as good.

Collaborative design it seems will always require a degree of both these top-down and bottom-up of coordinating processes. It seems that both approaches have their time and place, although traditionally within design, top-down approaches have been favored. With the growing complexity of collaborative design, reliance on top-down approaches alone may not be sufficient.

In the context of design systems however, the above arguments are greatly influenced by the intended degree of innovation in the design process. In routine design processes–ones in which the participants may have long experience, working within conceptual frameworks that are unlikely to change dramatically, top-down approaches can obviously provide useful support for designers. In routine design, the issue of design freedom is not normally relevant. Preconceived goals in such design situations are not really unwelcome constraints, but rather an essential feature of this type of design. It is only in design situations where innovation and creativity is desired, or when multiple perspectives must be incorporated, that bottom-up approaches to process management begin to seem appropriate. As argued above, such situations are fast becoming the norm in collaborative design.

3.3 **Complexity theory and process emergence**

Complex systems research has lately become a hugely popular scientific topic. This science concerns the general characteristics of decentralized, interacting systems. Often the thing that is most intriguing in such systems is their ability to self organize. Self-organization involves independent, but continuously interacting entities within these systems, forming patterns of process and structure, that are not intentional. That is, the intention of the independent entity is not a cause of the pattern, nor does it derive from the intentions of any other high-level entity that might have control over the entity. A process involving self-organization is called emergence.

The idea that these patterns arise without the intention of a 'creator', a 'designer', or some type of higher level cognitive entity, is according to Resnick (Resnick, 1994), a difficult concept for many people to comprehend.

The ideas of Darwin's theory of evolution are also related to those of complex systems, since they too propose that systems, that appear to be designed by a highly competent designer, have instead been formed by a complex, yet unintentional process. When Darwin's theory first came out in the middle of the last century, this aspect of it seemed to many to be completely preposterous (Mayr, 2000).

One of the common objections to it was how the simple mechanism of natural selection, which Darwin offered as the main mechanism of evolution, could possible result in the variety of life we find on earth. Natural selection involves the tendency of nature to over-produce the number of offspring, such that only a limited number - the most fit - will be available to survive, in the context of limited resources. In this way, species evolve by the selection of preferred, genetically based characteristics.

Of course, the idea of evolution is still highly controversial in many circles. However, within the scientific community it tends to be regarded now as scientific fact, rather than wild conjecture. It is also noteworthy to mention in the current design research context, that in the US a current euphemism for creationism is 'intelligent

The findings of complex systems research have been to applied to biological systems. Such research has augmented the mechanisms proposed by traditional Darwinism, or 'neo-Darwinism', to include that which has been discovered about self-organization in complex systems. In this view, not only through natural selection, but also through the tendency of interacting entities to learn to cooperate in mutually beneficial ways, leads to the formation of new patterns of organization, such as new species (Margulis, 1998).

The mechanisms of self-organization can be applied to biological systems, but are also relevant to any interacting system. If one takes the Darwinian position, that creation has not been created by a 'creator', and takes it to its logical conclusion, this means that mechanisms must be found to explain all patterns in non-biological systems as well (Eigen and Winkler-Oswatitsch, 1992). It appears that within the complex systems community, self-organization is the general process that is proposed to be fundamental in coming to understanding of patterns, and 'designs' in any complex, yet unintentional system.

Decentralized computational techniques inspired by the self-organizing processes of nature, have recently become well known. Computer algorithms that depend on large numbers of independent agents, inspired by the behaviors of ants and other social insects, and have been applied to a wide variety of problems, such as path-finding, sorting, clustering, and cooperative transport (Bonabeau and Théraulaz, 2000). The performance of some of these decentralized algorithms have been shown to be as effective as the best centralized alternatives, on common algorithmic problems such as the traveling salesman problem (Bonadeau, Dorigo, and Theraulaz, 1999).

3.3.1. Emergence in social processes

Ideas from complex systems research can also be applied, and have also arisen on their own, in study of social systems, in which emergence of processes and products, is also an important phenomenon. Schelling's work explores how the interaction of many independent agents - usually ordinary people - can form patterns they no person in particular seems to have intended, or could have intended. (Schelling, 1978). These interactions occur between agents and other agents, as well as between the environments in which the agents are situated.

Like complex systems research in general, such situations involve behaviors of an aggregate, such as a group of people attending a social event, and behaviors of the individuals that form that aggregate. In many cases, the behavior of the aggregate is not intuitively obvious, or cannot be extrapolated from, the behavior or the intentions of the individual. This is a very common theme in complex systems research.

Schelling talks about a mode of behavior, which he calls 'contingent' behavior. This behavior of individuals depends on what other individuals do. It is seen not as goal-directed, because it is emergent effect of an iterated, complex system. Such systems are not controlled by single agents who could conceivably form a goal.

A famous example that Schelling offers is a model of the self-organization of American neighborhoods into racially segregated neighborhoods. This he explains as an emergent effect of individual decisions to live within an area, which is populated predominantly by members of ones, own race. The segregation that results from such a model often exceeds the desire for segregation, of any one individual.

Axelrod's work continues in a similar vein to that of Schelling's (Axelrod, 1984) (Axelrod, 1997). In his research, Axelrod studies application of the Prisoner's Dilemma problem, which is a standard problem in game theory. Axelrod analyses emergent strategies that evolve when people play this game, not just a single time, but also multiple times. When played multiple times, the players begin to form a 'culture based on mutual influences' that depends on the sort of contingent behavior identified by Schelling. This kind of agent-based modeling of social systems is currently very popular. See for instance (Prietula, Carley, and Grasser, 1998) (Ferber, 1999).

4 DESIGN OF AN APPLICATION

This research has several goals that focus, on the coordination of collaborative and distributed design activity, on the study and representation of design processes, and on the study of design process emergence. The intent is not just to create theory, but also to construct a software prototype that demonstrates these ideas. This application has both a research and a practical orientation: to study and represent design processes, and well as to study ways of supporting designers in practice. This application is currently in implementation.

In order to make this kind of system work, what is required that its knowledge base be 'bootstrapped' from a state of no knowledge, or no content, to a state of greater knowledge. As the process content of the application grows, the tool should begin to provide more useful design support. Since the process knowledge base derives from the actions of only one designer, the process knowledge gathered could possibly be relevant only for an individual designer. What is perhaps lost in abandoning the top-down approach could perhaps be gained by access to useful process emergence. Such an idea remains a untested conjecture. Generality of the relevance of process knowledge acquired will not be assumed from the outset. However as a research vehicle, as well as a practical tool, the search for process patterns that appear to have relevance and generality for collaborative design in general, will be sought.

In the remainder of this paper some of the basic features of this application, that is designed to help coordinate collaborative design, is described.

4.1 **Application goals**

4.1.1. Help individual designers coordinate their work with their peers

Coordination is seen as a general process that can applied to many knowledge domains. The idea here is to help coordinate their processes, by helping them coordinate their process models, without the tool acquiring any semantic understanding of the content of these processes, process models.

4.1.2. Bring 'added value' to designers through coordination

In recording a designer's design process, something of value should be brought to the designer. That is, some motivation should be provided for the designer to use the application at all. Here this intended 'added value' is heightened design coordination between design collaborators.

4.1.3. Support individuals rather than organizations

Design of the application focuses completely on supporting individual designers, rather than an organization that might employ such a designer.

4.1.4. Support individual by recording their specific design processes

Derived from the idea that design support cannot be divorced from design process acquisition, record the design processes of individual designers. The intention is to acquire empirically based design process models, and provide an environment that encourages their acquisition.

4.1.5. Do not increase the cognitive or social burdens on designers, while attempting to support them

This involves keeping the tool simple and the cognitive and time demands on users low.

4.1.6. Do not require a group consensus before proceeding

This application is not intended to be one that requires first a costly consensus, or standardization effort on the part of designers, or managers to make it feasible. Instead, the approach is to design a tool that some might useful and supportive. Design support tools, especially ones that address such a personal matter to designers as design process, should not be imposed on designers, from the top-down.

4.1.7. Use simple message-based peer-to-peer communication

The systems will involve a peer-to-peer system in which all components will be the same, and will all therefore serve as both clients, and servers.

4.1.8. Two types of messages are required

- Process model content [represented as petri nets]
- Process model state [represented as petri net markings]

4.1.9. Process content is equivalent to a simple 'to-do' list

One of the simplest process models is the to-do list. Despite their simplicity, they can be extremely useful, especially if they are known to be relevant, and appropriate to a current situation. These process representations can contain information that must be defined by an author, as well as information that could be derived by the system. The intention here was to simplify as much as possible, the proposed content, and instead concentrate on general mechanisms that allow coordination of that content.

Information explicitly defined by an author:

- Tasks to be completed

Each task will have two attributes:

- The party expected to complete them, in the opinion of the person sending the message: a design stakeholder
- When they are expected to be done (in the opinion of the person sending the message)

Information implicitly defined, or derived by the system:

- The design project for which these tasks are relevant
- The design team which interacts on the design project

4.1.10. Petri nets/XML is the process representation

The proposed content is to be represented as simple [timed] petri nets.

Petri nets are a well-known process representation with several compelling advantages (Jensen, 1996): A clear graphical representation, the ability to handle both state and task process representations equally well. A syntax and semantics based on a small number of simple ideas. An ability to execute models dynamically. An ability to model true concurrency correctly. Such advantages make them particularly well suited to the modeling of, for instance, distributed algorithms.

4.1.11. Possible self organization through two simple processes

There are two processes proposed to provide the positive and negative feedback from users and their environment, which could conceivably allow self-organization to occur:

One is called completion analysis. This involves keeping track of whether the commitments have been kept. Therefore, the basic idea is that if you communicate your willingness to commit to a certain course of action, then you should be willing to go through with that commitment, unless some supportable reason comes up. Another is based on measured usage of models. Here the idea is promote process models to a higher level of visibility, and therefore to a greater chance of being reused, by measuring how much these models have been reused already. Reuse, then of good models becomes a positive reinforcement mechanism, in which their 'goodness' is rewarded.

5 CONCLUSION

In the context of creative design, process support that intends to truly supportive should not force a design methodology on designers. One form of design support would be to allow designers to do whatever they currently do, but provide means of coordinating their activities with those of their design collaborators. The most important factor leading towards design coordination is adequate communication. Many coordinating activities currently take place in design meetings, since they afford the opportunity to for design stakeholders to engage in face-to-face communication with their collaborators. In design meetings, designers and other stakeholders in the design process, not only communicate with each other, they also provide a forum for expressing social commitment, regarding the activities that each expects the other to perform in the future, and those that were expected to be performed in the past.

After giving such commitments to action in a social context, the social costs, in cases of non-performance of a promised activity, tends to increase to the offending party. Giving social commitment in a social context is one way of reducing the unpredictability and complexity of collaborative design, by increasing the likelihood that specific activities, and specific design directions, will be pursued in the future. One problem, however with design meetings, despite the important social and technical functions they serve, is that they are often expensive to hold, and can often exclude stakeholders in the design process whose presence could be helpful in coming to well-balanced solutions. A simple distributed groupware system could support some of the advantages of design meetings, such as communication of process plans and social commitments to these plans, but reduce the cost and effort of such communication.

The distributed approach to design support is chosen over the centralized one, because centralized systems often impose, wittingly or unwittingly, a prescriptive design methodology on designers. Designers may, or may not, find such a prescription helpful or supportive of their practice. Indeed what constitutes the practice of a creative designer, in changing times, is itself a moving target. This is thought to be especially true in creative design, whose processes, by definition, tend to be less predictable than those designing in routine situations, in which process regularities may be more apparent to the participants. As well, centralized systems by the simple fact that they are centralized, usually assume that a 'center' exists in the design process. Such an assumption may not be appropriate in many collaborative design situations where control hierarchies either are flattened, or can change quickly, and unpredictably.

6 REFERENCES

Akin, Ö., and C. Lin. (1995). Design protocol data and novel design decisions. *Design Studies*. 16. 211-236.

Axelrod, R. (1984). *The Evolution of Cooperation*. New York, NY: Basic Books.

Axelrod, R. (1997). *The Complexity of Cooperation: Agent-Based Models of Competition and Collaboration*. Princeton, NJ: Princeton University Press.

Bonabeau, E., and G. Théraulaz. (2000). Swarm Smarts. *Scientific American,* March 2000, 72-79.

Bonadeau, E., M. Dorigo, and G. Theraulaz. (1999). *Swarm Intelligence: From natural to artificial systems.* New York, NY: Oxford University Press.

Bucciarelli, L. (1994). *Designing Engineers*. Cambridge, MA: The MIT Press.

Clark, H. H. (1996). Using Language. Cambridge, UK: Cambridge University Press.

Clarke, I. (1999). *A Distributed Decentralized Information Storage and Retrieval System* Project Report, Supervisor: Dr. Chris Mellish. Edinburgh, UK: Division of Informatics, University of Edinburgh.

Cross, N. (1993). Science and Design Methodology: A Review. Research in Engineering Design. Vol.5. 63-69.

Cross, N., H. Christiaans, and K. Dorst. (1996). Introduction: The Delft Protocols Workshop. In N. Cross & H. Christiaans & K. Dorst Eds. *Analysing Design Activity.* Chichester, UK: John Wiley & Sons.

Cross, N., and A. C. Cross. (1996). Observations of Teamwork and Social Processes in Design. In N. Cross & H. Christiaans & K. Dorst Eds. *Analysing Design Activity.* Chichester, UK: John Wiley & Sons.

Darke, J. (1984). The Primary Generator and the Design Process. In N. Cross Ed. *Developments in Design Methodology*. New York, NY: John Wiley & Sons.

Dym, C., and R. Levitt. (1991). *Knowledge-Based Systems in Engineering*. New York, NY: McGraw-Hill.

Eigen, M., and R. Winkler-Oswatitsch. (1992). *Steps towards Life: A perspective on evolution* P. Wooley, Trans. Oxford: Oxford University Press.

Ericsson, K., and H. Simon. (1980). Verbal Reports as Data. *Psychological Review.* Vol.87. No.3. 215-250.

Evan, W. (1993). *Organizational Theory: Research and Design*. New York, NY: Macmillan.

Ferber, J. (1999). *Multi-agent Systems: An introduction to distributed artificial intelligence*. Harlow: Addison-Wesley.

Guindon, R. (1990). Designing the Design Process: Exploiting Opportunistic Thoughts. *Human-Computer Interaction.* Vol.5. 305-344.

Jensen, K. (1996). *Coloured Petri Nets: Basic Concepts.* 2nd ed. Vol. 1. Berlin: Springer Verlag.

Klein, M. (1998). Coordination Science: Challenges and Directions. In W. Conen & G. Neumann Eds. *Coordination Technology for Collaborative Applications*. 161-176. Berlin: Springer Verlag.

Malone, T., and K. Crowston. (1992). What is Coordination Theory and How Can It Help Design Cooperative Work Systems? In D. Marca & G. Bock Eds. *Groupware: Software for Computer-Supported Cooperative Work*. Los Alamitos, CA: IEEE Computer Society Press.

Margulis, L. (1998). *Symbiotic Planet*. New York, NY: Basic Books.

Mayr, E. (2000). Darwin's Influence on Modern Thought. *Scientific American,* July 2000, 79-83.

Medina-Mora, R., T. Winograd, R. Flores, and F. Flores. (1992). The Action Workflow Approach to Workflow Management Technology. Proceedings of CSCW '92. Toronto.

Prietula, M., K. Carley, and L. Grasser Eds. (1998). *Simulating Organizations: Computational Models of Institutions and Groups*. Menlo Park, CA: AAAI Press.

Resnick, M. (1994). *Turtles, Termites, and Traffic Jams: Explorations in massively parallel microworlds*. Cambridge, MA: The MIT Press.

Roozenburg, N., and N. Cross. (1991). Models of the design process: integrating across the disciplines. *Design Studies.* Vol.12. No.4. 215-220.

Schelling, T. (1978). *Micromotives and Macrobehavior*. New York, NY: Norton.

Schön, D. (1983). *The Reflective Practitioner: How Professionals Think in Action*. New York, NY: Basic Books.

Searle, J. (1991). Response: Meaning, Intentionality, and Speech Acts. In E. Lepore & R. v. Gulick Eds. *John Searle and his Critics*. Oxford, UK: Blackwell.

Software Engineering Institute, C. M. U. (1995). *The Capability Maturity Model: Guidelines for Improving the Software Process*. Reading, MA: Addison-Wesley.

Whitney, D. (1990). Designing the Design Process. *Research in Engineering Design.* Vol.2. 3-13.

Winograd, T., and F. Flores. (1987). *Understanding Computers and Cognition: A new foundation for design*. Reading, MA: Addison-Wesley.