Artificial Intelligence for Engineering Design, Analysis and Manufacturing (2007), *21*, 295–308. Printed in the USA.

DOI: 10.1017/S0890060407000297

An ontology of situated design teams

JOHN S. GERO[1] AND UDO KANNENGIESSER[2]

[1]Krasnow Institute for Advanced Study and Department of Computer Science, George Mason University, Fairfax, Virginia, USA
[2]NICTA, Alexandria, Australia

(RECEIVED May 25, 2006; ACCEPTED October 5, 2006)

Abstract

This paper presents an ontological framework for situated design teams in which the team is both the subject and the object of designing. Team designing is modeled using the set of processes provided by the situated function–behavior–structure framework. This is a formal basis for understanding the drivers for change in the product to be designed and in the design team. We specifically focus on changes in a team's structure that emerge from interactions among individual team members and subteams.

Keywords: Design Teams; Function–Behavior–Structure Framework; Situatedness; Team Interaction

1. INTRODUCTION

Situated designing is a paradigm that has received increasing attention in the design research community. It has been investigated by means of protocol studies, computational simulations, and ontological frameworks. The notion of "situatedness" is used to describe how design processes lead to different results depending on the unique experience of the designer. This experience is formed as a result of the designer's interactions with representations of the current design process and previous design processes. The central role of interaction in this view of designing allows capturing the potential for changes to occur both in the course of the ongoing design process and in the designer's experience.

The majority of research in situated designing has focused on studying individual designers and their interactions with external design representations. However, designed products are rarely the result of an isolated activity of one individual person. Most design processes are carried out by teams, ranging from just a handful of designers to large organizations involving hundreds of domain experts. Interactions between different stakeholders of a collaborative design project strongly influence the course and outcomes of the design. In turn, the design team itself changes as a result of its interactions, creating new organizational knowledge (Nonaka, 1994) and potentially new organizational structures. Work in distributed artificial intelligence has viewed emergent team reorganization as an instance of (self-)designing (Corkill & Lesser, 1983).

This paper presents an ontological framework for situated design teams in which the team is both the subject and the object of designing. This facilitates understanding of the effects and mechanisms of situatedness in team designing and provides a basis for computational modeling of situated design teams.

2. REPRESENTING SITUATED DESIGNING

2.1. The function–structure–behavior (FBS) ontology

Gero's (1990) FBS ontology provides three high-level categories for the properties of an object:

1. The *function* of an object is defined as its teleology, that is, "what the object is for."
2. The *behavior* of an object is defined as the attributes that are derived or expected to be derived from its structure, that is, "what the object does."
3. The *structure* of an object is defined as its components and their relationships, that is, "what the object consists of."

Humans construct connections between function, behavior, and structure through experience and through the development of causal models based on interactions with the object. Specifically, function is ascribed to behavior by establishing a teleological connection between the human's

Reprint requests to: Udo Kannengiesser, NICTA, Locked Bag 9013, Alexandria, NSW 1435, Australia. E-mail: udo.kannengiesser@nicta.com.au

goals and observable or measurable effects of the object. Behavior is causally connected to structure, that is, it can be derived from structure using physical laws or heuristics. There is no direct connection between function and structure (de Kleer & Brown, 1984).

The generality of the FBS ontology allows for multiple views of the same object. This enables the construction of different models depending on their purpose. For example, an architectural view of a building object includes different FBS properties than a structural engineering view. This is most striking for the building's structure: architects typically view this structure as a configuration of spaces, whereas engineers often prefer a disjoint view based on floors and columns.

Multiple views can also be constructed depending on the required level of aggregation. This allows modeling objects as assemblies composed of subassemblies and individual parts. Each of these components can again contain other subassemblies or parts. No matter which level of aggregation is required, the FBS ontology can be applied.

Gero (1990) has used the FBS ontology as the basis of a framework that describes designing as a set of eight fundamental processes (Fig. 1):

1. *Formulation* (process 1 labeled in Fig. 1) transforms the design requirements, expressed in function, into behavior that is expected to enable this function.
2. *Synthesis* (process 2) transforms the expected behavior into a solution structure that is intended to exhibit this desired behavior.
3. *Analysis* (process 3) derives the "actual" behavior from the synthesized structure.
4. *Evaluation* (process 4) compares the behavior derived from structure with the expected behavior to prepare the decision if the design solution is to be accepted.

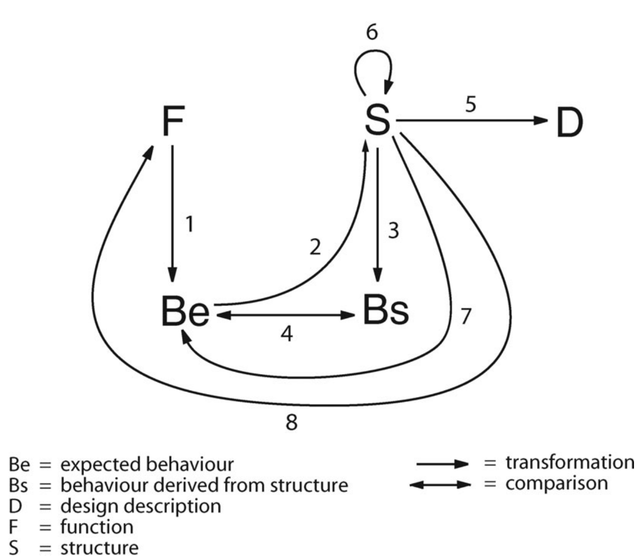


Fig. 1. The function–behavior–structure framework (Gero, 1990).

5. *Documentation* (process 5) produces the design description for constructing or manufacturing the product.
6. *Reformulation type 1* (process 6) addresses changes in the design state space in terms of structure variables or ranges of values for them if the actual behavior is evaluated to be unsatisfactory.
7. *Reformulation type 2* (process 7) addresses changes in the design state space in terms of behavior variables or ranges of values for them if the actual behavior is evaluated to be unsatisfactory.
8. *Reformulation type 3* (process 8) addresses changes in the design state space in terms of function variables or ranges of values for them if the actual behavior is evaluated to be unsatisfactory.

2.2. Situatedness

Designing is an activity during which designers perform actions to change their environment. By observing and interpreting the results of their actions, they then decide on new actions to be executed on the environment. This means that the designers' concepts may change according to what they are "seeing," which itself is a function of what they have done. One may speak of an "interaction of making and seeing" (Schön & Wiggins, 1992). This interaction between the designer and the environment strongly determines the course of designing. This idea is called situatedness, whose foundational concepts go back to the work of Dewey (1981) and Bartlett (1932).

In experimental studies of designers, phenomena related to the use of sketches, which support this idea, have been reported. Schön and Wiggins (1992) found that designers use their sketches not only as an external memory, but also as a means to reinterpret what they have drawn, thus leading the design in a new direction. Suwa et al. (1999) noted, in studying designers, a correlation of unexpected discoveries in sketches with the invention of new issues or requirements during the design process. They concluded that "sketches serve as a physical setting in which design thoughts are constructed on the fly in a situated way."

Gero and Fujii (2000) have developed a framework using situated cognition, which describes the designer's interpretation of their environment as interconnected sensation, perception, and conception processes. Each of them consists of two parallel processes that interact with each other: A *push process* (or data-driven process), where the production of an internal representation is driven ("pushed") by the environment, and a *pull process* (or expectation-driven process), where the interpretation is driven ("pulled") by some of the designer's current concepts, which has the effect that the interpreted environment is biased to match the current expectations.

The environment that is interpreted can be external or internal to the agent. The situated interpretation of the internal environment accounts for the notion of constructive memory. The relevance of this notion in the area of design research

https://doi.org/10.1017/S0890060407000297 Published online by Cambridge University Press

has been shown by Gero (1999). Constructive memory is best exemplified by a paraphrase of Dewey by Clancey (1997): "Sequences of acts are composed such that subsequent experiences categorize and hence give meaning to what was experienced before." The implication of this is that memory is not laid down and fixed at the time of the original sensate experience but is a function of what comes later as well. Memories can therefore be viewed as being constructed in response to a specific demand, based on the original experience as well as the situation pertaining at the time of the demand for this memory. Therefore, everything that has happened since the original experience determines the result of memory construction. Each memory, after it has been constructed, becomes part of the existing knowledge (and becomes part of a new situation) and is now available to be used later, when new demands require the construction of further memories. These new memories can be viewed as new interpretations of the augmented knowledge. Figure 2 shows the idea of constructive memory graphically.

The advantage of constructive memory is that the same external demand for a memory can potentially produce a different result at different times, as newly acquired experiences may take part in the construction of that memory. Constructive memory can be seen as the capability to integrate new experiences by using them in constructing new memories. As a result, knowledge "wires itself up" based on the specific experiences it has had, rather than being fixed, and actions based on that knowledge can be altered in the light of new experiences.

Situated designing uses first-person knowledge grounded in the designer's interactions with their environment (Bickhard & Campbell, 1996; Clancey, 1997; Ziemke, 1999; Smith & Gero, 2005). This is in contrast to static approaches that attempt to encode all relevant design knowledge prior to its use. Evidence in support of first-person knowledge is provided by different designers producing different designs for the same set of requirements, and the same designer is likely to produce different designs at later times for the same requirements. This is a result of the designer acquiring new knowledge while interacting with their environment between the two times.

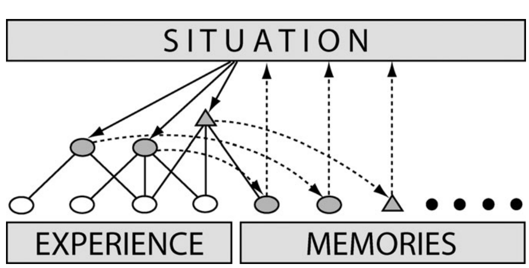


Fig. 2. The original experiences (unshaded ellipses) and the situation are used to construct memories of the experience (shaded ellipses), then these memories are added as experiences and may be used later to produce further new memories (shaded triangles) in conjunction with later situations and so on.

Gero and Kannengiesser (2004*a*) have modeled situatedness as the interaction of three worlds, each of which can bring about changes in any of the other worlds. The three worlds include the observer's external world, interpreted world, and expected world (Fig. 3a). The definition of each world implies the existence of an individual designer or design agent:

1. The *external world* is the world that is composed of representations outside the design agent.
2. The *interpreted world* is the world that is built up inside the design agent in terms of sensory experiences, percepts, and concepts. It is the internal representation of that part of the external world that the design agent interacts with.
3. The *expected world* is the world imagined actions of the design agent will produce. It is the environment in which the effects of actions are predicted according to current goals and interpretations of the current state of the world.

These three worlds are linked together by three classes of connections. *Interpretation* transforms variables that are sensed in the external world into the interpretations of sensory experiences, percepts, and concepts that compose the interpreted world. *Focusing* takes some aspects of the interpreted world, uses them as goals in the expected world, and suggests actions, which, if executed in the external world should produce states that reach the goals. *Action* is an effect that brings about a change in the external world according to the goals in the expected world.

Figure 3b presents a specialized form of this view with the design agent (as the internal world) located within the external world and placing general classes of design representations into the resultant "onion" model. The set of expected design representations (Xe^i) corresponds to the notion of a design state space, that is, the state space of all possible designs that satisfy the set of requirements. This state space can be modified during the process of designing by transferring new interpreted design representations (X^i) into the expected world and/or transferring some of the expected design representations (Xe^i) out of the expected world. This leads to changes in external design representations (X^e), which may then be used as a basis for reinterpretation changing the interpreted world. Novel interpreted design representations (X^i) may also be the result of constructive memory, which can be viewed as a process of interaction among design representations within the interpreted world rather than across the interpreted and the external world. Both interpretation and constructive memory are viewed as push–pull processes.

2.3. The situated FBS framework

Gero and Kannengiesser (2004*a*) have used the model of situatedness presented in Figure 3 as a basis for integrating the notion of situatedness into Gero's (1990) original FBS

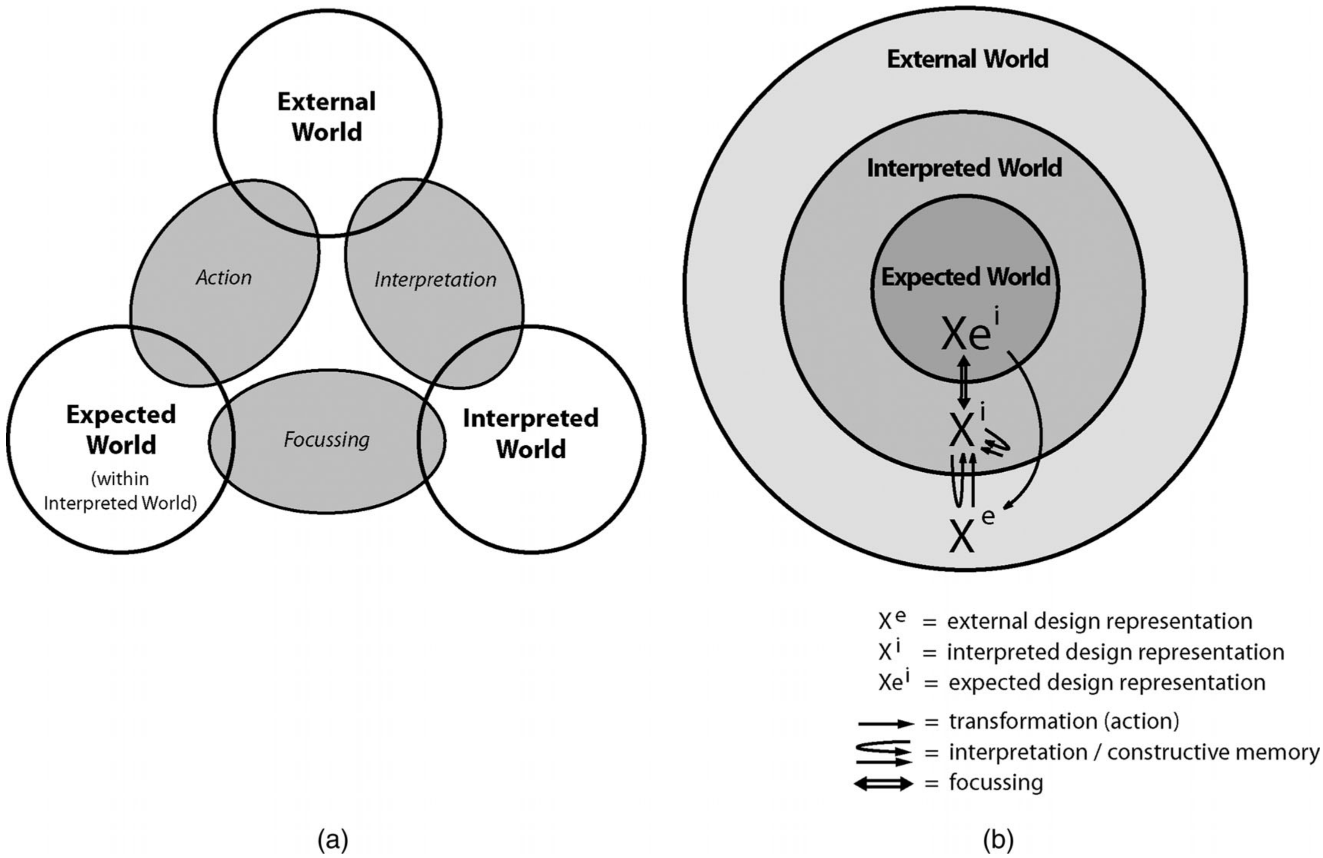


Fig. 3. Situatedness as the interaction of three worlds: (a) general model and (b) specialized model for design representations (after Gero & Kannengiesser, 2004*a*).

framework, thus forming the situated FBS framework (Fig. 4). This framework has the capacity to describe how interactions between the design agent's current goals, interpretations, and environment can lead to modifications of the function, behavior, and structure of the design object. The eight fundamental processes of the original FBS framework can now be represented in a more detailed way that includes their situatedness.

1. *Formulation:* This consists of processes 1–10 (see Fig. 4). It includes interpretation of explicit requirements given to the design agent in the external world as function, behavior, and structure, via processes 1, 2, and 3. For example, the requirements for designing a window may refer to the functions "enhancing winter solar gain" and "controlling noise," the behavior "thermal conduction," and structure constraints on the variables "glazing length" and "glazing height." These requirements are complemented by implicit requirements generated from within the agent, namely, by constructive memory (processes 4, 5, and 6). In the window example, implicit requirements may include the function "proving view," the behavior "light transmission," and the structure variable "type of coating." Focusing transfers a subset of the required function, behavior, and structure into the expected world (processes 7, 8, and 9). Additional behavior is constructed from function via process 10, which is generally viewed as the main concern of requirements engineering. For example, the function "enhancing winter solar gain" is transformed into the behavior "direct solar gain."
2. *Synthesis:* Synthesis consists of process 11 to generate a structure that is expected to meet the required behavior and the externalization of that structure via process 12. In the window example, synthesis generates values for the formulated structure variables "glazing length," "glazing height," and "type of coating." This design candidate can be externalized in form of iconic or symbolic representations.
3. *Analysis:* This consists of interpretation of the external structure (process 13) and the derivation of behavior from that structure (process 14). Examples of analyses in window designing include structural and thermal analysis.
4. *Evaluation:* Evaluation consists of process 15 that includes a comparison of the actual and the expected

https://doi.org/10.1017/S0890060407000297 Published online by Cambridge University Press

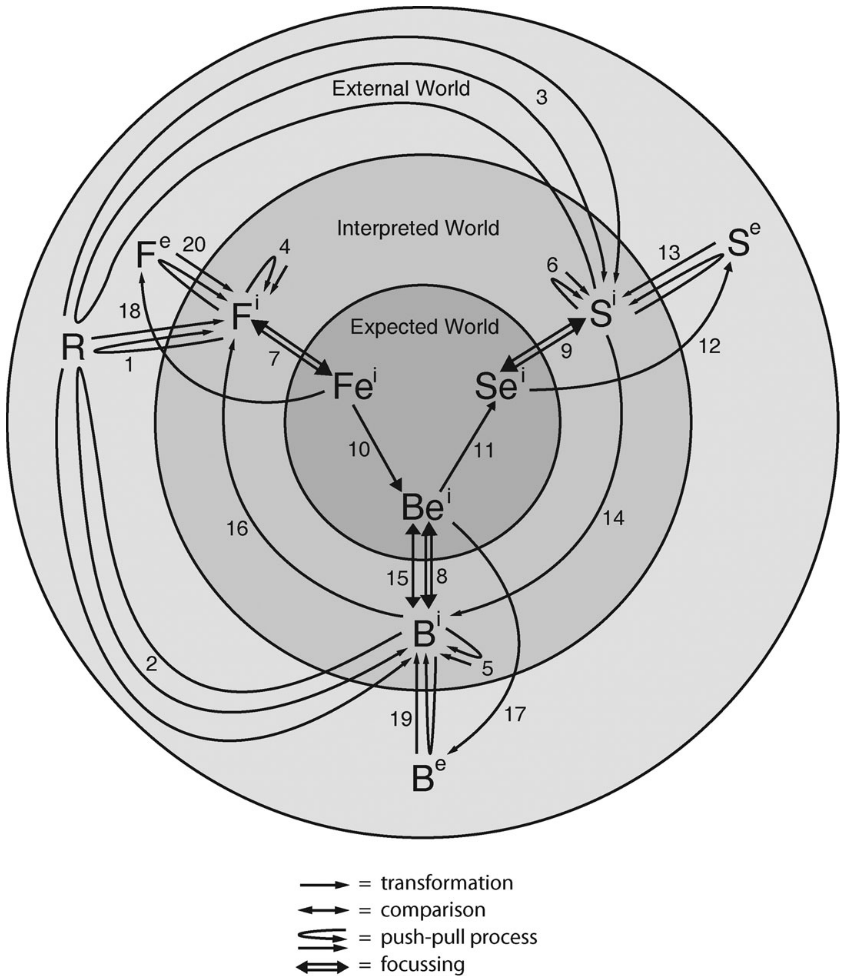


Fig. 4. The situated function–behavior–structure framework (Gero & Kannengiesser, 2004*a*).

behavior. For example, a window can be evaluated by comparing the expected value with the derived ("actual") value of the window's thermal conduction.

5. *Documentation:* Documentation produces an external representation of the final design solution for purposes of communicating that solution in terms of structure (process 12), and, optionally, behavior (process 17) and function (process 18). Common products of physical designs such as windows are computer-aided design models and component lists.
6. *Reformulation type 1:* This consists of focusing on different structures (process 9). Precursors of this process are the interpretation of external structure (process 13), constructive memory of structure (process 6), or the interpretation of new requirements on structure (process 3). In the window example, reformulation type 1 may introduce the new structure variable "angle," resulting in a nonorthogonal relationship between "glazing length" and "glazing height."
7. *Reformulation type 2:* Type 2 consists of focusing on different behaviors (process 8). Precursors of this process are the derivation of behavior from structure (process 14), the interpretation of external behavior (process 19), constructive memory of behavior (process 5), or the interpretation of new requirements on behavior (process 2). In the window example, reformulation type 2 may change the window's opening mechanism by substituting the behavior "rotating" by "sliding."
8. *Reformulation type 3:* This consists of focusing on different functions (process 7). Precursors of this process are the ascription of function to behavior (process 16), the interpretation of external function (process 20), constructive memory of function (process 4), or the interpretation of new requirements on function (process 1). In the window example, reformulation type 3 may introduce the function "providing access into the building," which points toward a combined "window-and-door" design.

3. SITUATED TEAM DESIGNING

As described in Section 1, the notion of situated team designing implies the process of designing not only the product but

https://doi.org/10.1017/S0890060407000297 Published online by Cambridge University Press

also the team. We will refer to the former as *p-designing* (with the team as the subject of designing) and to the latter as *t-designing* (with the team as the object of designing).

3.1. P-Designing

The situated FBS framework has been used to describe the activities carried out by an individual designer. It implied the existence of an agent embodying the interpreted and expected worlds. Although this assumption has been helpful for initial understanding, it obstructs the view of the situated FBS framework as an ontology that can represent all instances of designing independent of their embodiment. Such a view can account for designing both by individual designers and by teams of designers.

The view of the situated FBS framework as an ontology focuses on classes of design activities rather than on classes of design generators (i.e., agents). This enables independence of the framework not only from the particular object to be designed but also from the particular subject generating the design. The model of situatedness as three interacting worlds is viewed as an ontology, allowing for multiple views of each of the three worlds and their interconnections. Two basic views can be distinguished when applying the three-world model to teams of agents.

The "individualist" view maps all ontological elements in the model onto the states and activities attributed to individual members of the team. One implication of this view is that the number of processes required to model p-designing in the situated FBS framework increases with the number of team members. The communication necessary among the members to achieve coherent design solutions further increases the number of processes, primarily by adding action and interpretation processes for message exchange. Emergent, global notions such as a common design state space of the team are not considered in this view.

The "social" view adopts a higher degree of granularity, regarding coordinated states and processes as elementary within the situated FBS framework. These states and processes are composites of individual states and processes attributed to a set of team members or subteams. The "social" view can therefore be seen as identifying the team as a "superagent," in a way reminiscent of Minsky's (1985) *The Society of Mind*. This view requires further elaboration by reinterpreting the three-world model in the context of an assumed superagent:

- The *external world of a design team*, adopting the "social" view, is the world that is composed of representations outside the team. It includes representations that are used for communication with the team itself or other agents or groups of agents. Examples for external representations that a team commonly deals with are the requirements given by the customer, the design descriptions produced for the manufacturer, and the project reports written for the team's supervisor.
- The *interpreted world of a design team*, adopting the "social" view, is the world that is built up inside the team in terms of sensory experiences, percepts, and concepts of individual team members or subteams, and communicative actions among the individual team members or subteams for purposes of coordination. This idea draws on the notion of transactive memory (Wegner, 1986), which proposes a view of the "group mind" based on interactions between multiple information sources embodied in different individual agents.
- The *expected world of a design team*, adopting the "social" view, is the world that the imagined actions of the team will produce. It is the environment in which the effects of actions are predicted according to the team's current joint goals and interpretations. Expected representations may be specified explicitly via some interagent communication medium (e.g., e-mails, shared project database, paper-based documents, etc.) or implicitly based on tacit agreement among the team.
- *Interpretation by a design team*, adopting the "social" view, transforms variables, which are sensed in the external world into sensory experiences, percepts, and concepts of individual team members in compliance with Gero and Fujii's (2000) framework. This process may involve interactions among team members with the purpose of eliminating ambiguities or differences between individual interpretations.
- *Focusing by a design team*, adopting the "social" view, transfers aspects of the interpreted world into the expected world, producing a set of joint goals and actions. This process requires some form of decision mechanism to prevent differences in the design preferences of individual team members from affecting the consistency of the overall team's design state space.
- *Action by a design team*, adopting the "social" view, is an effect that brings about a change in the external world according to the team's joint goals in the expected world. This process includes a notion of joint commitment of the team (Cohen & Levesque, 1991) rather than only the individual commitment of the team members ultimately executing the action.
- *Constructive memory of a design team*, adopting the "social" view, includes interactions between team members accessing transactive memory. It combines constructive memory processes of individual team members with the interpretations, hypothesizing, and actions involved when engaging in transactive memory processes.

3.2. T-Designing

Applying the situated FBS framework to t-designing requires a representation of teams in terms of function, behavior, and structure. An obvious example for a team function is to carry out the assigned p-design task. Typical examples of team behavior are the time to produce a result and labor cost. The structure of a team encompasses individual designers or groups of designers and their relationships instantiated through individual interactions and flows of information (Galbraith, 1977).

Who is the subject in t-designing? It can be any entity that, at any degree of granularity, has the capacity to interpret, construct memories, focus, and act with respect to different representations of a t-design. In Section 3.1 we have seen that both an individual and a team can instantiate this entity. An example of the former is a project manager who configures and reconfigures a team according to a given task to be performed by that team. However, we want to focus on the case of a team being the t-design generator, that is, a team designing itself. In other words, this is the case where the team that is the design object is identical with the team that is the design generator.

We illustrate the eight fundamental processes in the situated FBS framework for our case of t-designing, adopting a "social" view of the team. Take the example of a small design team that includes a team leader using a cooperative mode of leadership.

1. *Formulation:* The team leader is given a set of requirements from a supervisor. These requirements are interpreted as the following:

   - function (F^i, via process 1 labeled in Fig. 4), for example, a set of p-design tasks to be carried out such as designing the engine and the transmission system of a car;
   - behavior (B^i, via process 2) in terms of the performance expected from the team, including time and deliverables; and
   - structure (S^i, via process 3) in terms of some of the members of the team and their (subordinate) relationship to the team leader.

   The team leader augments this set of requirements via constructive memory that is instantiated either as the team leader's individual memory or as the group's (transactive) memory constructed in an initial team meeting and later. Possible implicit requirements originating from constructive memory are related to the following:

   - function (F^i, via process 4), for example, more detailed tasks to be carried out such as performing thermodynamic analyses or process-related goals such as conformance to quality standards;
   - behavior (B^i, via process 5), for example, the time required to achieve specific subtasks; and
   - structure (S^i, via process 6), for example, a small subteam of two or three members that collaborated successfully in past design projects, or potential additional members of the team.

   A subset of all explicit and implicit requirements is then transferred into the t-design state space (processes 7, 8, and 9), by agreement of all team members or by decision of the team leader. Additional expected behaviors are then specified via process 10.

2. *Synthesis:* After the team has been formulated, its specific expected structure (Se^i) is instantiated (process 11) and externalized (process 12) through interactions that conform to the relationships between team members. For example, progress reports produced by individual team members for the team leader are externalized instantiations of the supervision relationship.
3. *Analysis:* Interpretation of externalized team structure (process 13) and derivation of interpreted behavior (B^i) from the interpreted structure (S^i, via process 14) is carried out to monitor the performance of the design team.
4. *Evaluation:* The actual behavior is compared to the expected behavior (Be^i) to provide a basis for potential control actions to be taken by the team leader.
5. *Documentation:* An external representation of the team's structure (S^e, via process 12) is rarely required. This is in contradistinction to the documentation when designing physical objects. More frequent are documentations of team performance (B^e, via process 17) and functions (F^e, via process 18), as a basis for evaluating the team at the end of the design project.
6. *Reformulation type 1:* The team leader (and/or the whole team) might find that, to decrease the time required, additional team members are needed. Another example is the modification of relationships within the team. These are structure changes that are modeled by focusing (process 9) that alters the t-design state space. Drivers for this process are the following:

   - interpretation (process 13), for example, from advice given by the team leader's supervisor on team building;
   - constructive memory (process 6), for example, by becoming aware of the existence of a successful informal group within the team that works more effectively than a formally appointed subteam; and
   - new requirements (process 3), for example, changes in the human resources available, passed on to the team leader by their supervisor.

7. *Reformulation type 2:* The team leader (and/or the whole team) might find that relaxing constraints on the team's working speed is required to improve product quality (process 8). Drivers for this process are the following:

   - derivation of behavior from structure (process 14)—one way of instantiating this process is by behavior analogy based on structure similarity with a source t-design, for example, similarities in the distribution of specialist knowledge across the team may introduce new behaviors into the target t-design;
   - interpretation (process 19), for example, from a documentation of past team performance;
   - constructive memory (process 5)—here the team may internally generate modifications of its own

https://doi.org/10.1017/S0890060407000297 Published online by Cambridge University Press

behavior, for example, a group or subteam may realize that current deadlines cannot realistically be met, and this group may then communicate a suggestion for a modified schedule to the team leader or the whole team; and

- new requirements (process 2), for example, new time constraints imposed on the team by their supervisor.

8. *Reformulation type 3:* The team leader (and/or the whole team) might find that functions related to domain tasks (i.e., p-design tasks) might change (process 7). For example, the team might drop the function of designing the transmission system and concentrate exclusively on designing the engine. Another example is the self-assignment of additional functions, such as including the design of the exhaust system with the design of the engine. Drivers of reformulated function are

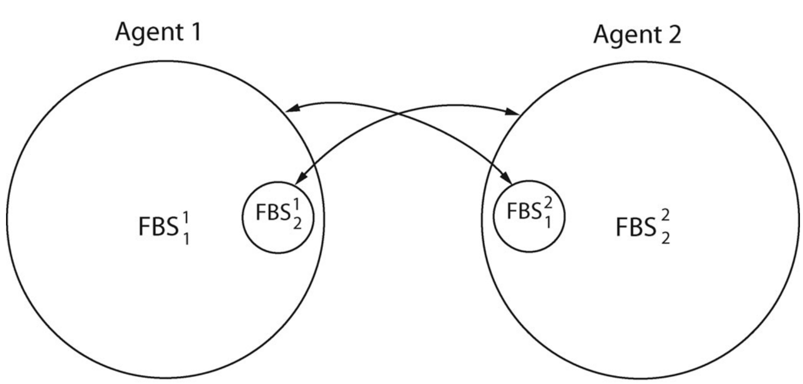
   - ascription of function to behavior (process 16), for example, to the team's behavior of introducing new structural constraints for the exhaust system (because of the particular engine layout), which may then produce the new team function "to adapt the exhaust system to the engine";
   - interpretation (process 20), for example, from a request by management to accept an extended list of p-design components to be designed;
   - constructive memory (process 4), for example, by the team leader realizing the availability of expertise in the team that could be used to extend current design responsibilities; and
   - new requirements (process 1), for example, from the customer asking for a more comprehensive set of p-design tasks.

4. DRIVERS FOR CHANGE IN SITUATED DESIGN TEAMS

In this section we provide an understanding of how new team structures can emerge from within the team. In our situated FBS framework applied to t-designing, this concerns process 6. In particular, we want to focus on informal interactions within the team, which have the potential to provide the basis for the formation of new, formal relationships.

4.1. Situated interaction between individual team members

All interactions within a team are ultimately carried out by individual team members. For an interaction to be successful there needs to be a common ground (Clark, 1996) among all individuals involved in that interaction. Common ground is knowledge that is shared and known to be shared. Specifically, it requires that both participants construct adequate internal representations of each other to evaluate the existence of sufficient common ground for the current purposes of the interaction. Gero and Kannengiesser (2004*b*) have used the FBS ontology as a basis for internal representations and have shown how this ontology facilitates the evaluation of a common ground. Function, behavior, and structure are here applied to model individual agents. For example, a team member may have a function to prepare cost evaluation, a behavior to deliver the total cost of a building, and a structure that includes that team member's expert knowledge necessary to produce this behavior. Gero and Kannengiesser's (2004*b*) model of common ground is shown in Figure 5. Here the knowledge of two agents is represented as the set of FBS models they have constructed of each other (including of



FBS^1_1 = FBS models of agent 1 constructed by agent 1
FBS^1_2 = FBS models of agent 2 constructed by agent 1
FBS^2_2 = FBS models of agent 2 constructed by agent 2
FBS^2_1 = FBS models of agent 1 constructed by agent 2

Fig. 5. Pairs of consistent function–behavior–structure models that establish the common ground of two agents (Gero & Kannengiesser, 2004*b*).

https://doi.org/10.1017/S0890060407000297 Published online by Cambridge University Press

themselves). Common ground then encompasses those parts of an agent's FBS model that are consistent with the corresponding FBS model constructed by the other agent.

A sufficient amount of structure, that is, the knowledge structure including an agent's ontologies, is critical in the construction of FBS models to reach common ground in communication. An agent can generally use two sources of information to access another agent's S. The first one includes those parts of S that the other agent makes directly available by communicating them. The second one includes generalizations over a set of previous experiences with other, similar agents. Cues for constructing these generalizations are often provided by observations of the other agent's behavior. Usually both sources of information are employed, with generalizations typically providing default assumptions when only incomplete information is available from direct communication. A large part of generalizations are constructed from the agent's FBS model of itself.

Figure 6 illustrates these effects for an agent 0 having constructed FBS models of four other agents (1, 2, 3, and 4). As the differently sized FBS models in the figure indicate, some agents (1 and 2) are better known (grounded) than others (3 and 4), and the best-known agent is certainly agent 0 itself. When the agent wants to interact with one of the other agents but has too little knowledge about that agent (here 4) to establish sufficient common ground for this interaction, it complements the existing FBS model with assumptions reflecting its generalized knowledge about similar agents. This generalized knowledge is derived mainly from those instances the agent 0 is most familiar with, as indicated by the different weights of the arrows in Figure 6, which principally includes the agent 0 itself. When a new, previously unknown agent 5 enters agent 0's team, the generalized knowledge may still suffice to construct an adequate FBS model of that agent using the generalized knowledge about function, behavior, and structure individually and their relationships. If there is a conflict between the generalized knowledge and the interactions with a specific agent, then a specialized FBS view of that agent needs to be constructed.

Most work on the use of common ground in design (Gero & Kannengiesser, 2004*b*; Kannengiesser & Gero, 2007) focuses on the structure part of the agents' FBS models to address issues of interoperability when interactions involve multiple domain ontologies. This aspect is important, because it determines the feasibility of interaction through constraining the agents' structure and thus their behavior in

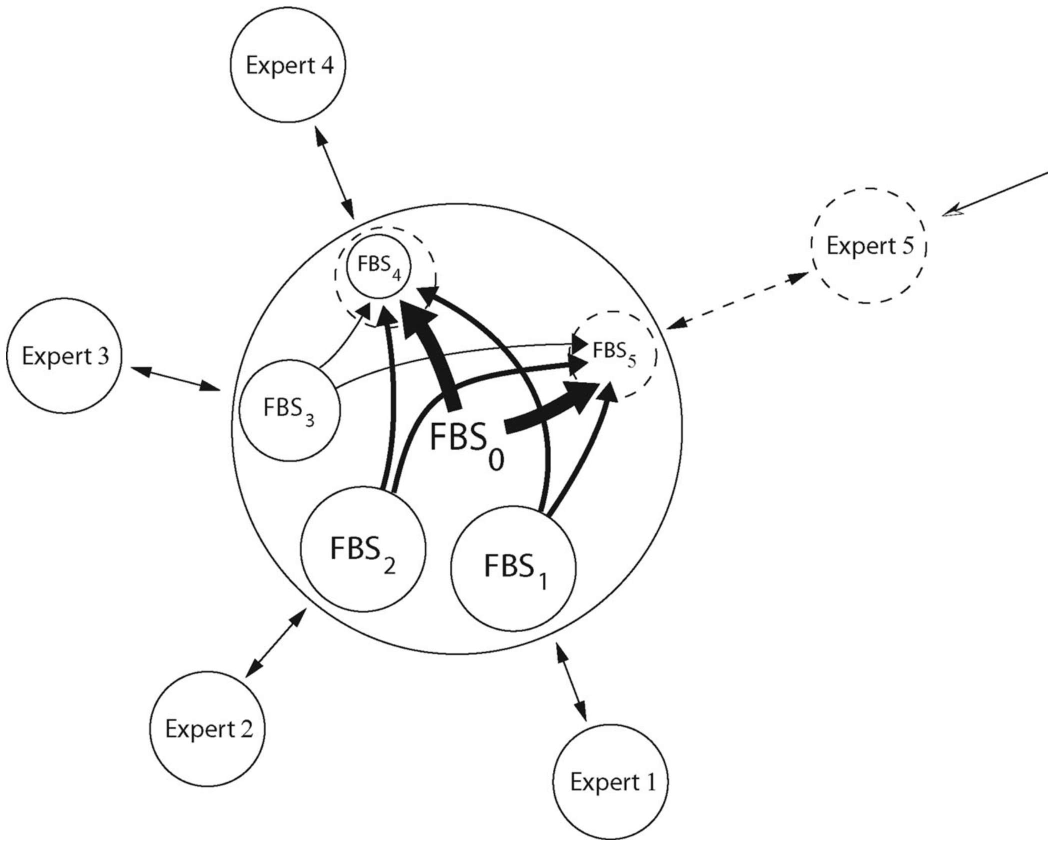


Fig. 6. New FBS models are constructed using generalizations of previously constructed function–behavior–structure (FBS) models. The size of the circle for each FBS is an indication of the amount of grounding of this FBS model of the other agent. The width of the arrows is an indication of the confidence of the potential applicability of the originating FBS model in constructing or supporting the FBS model of a new agent.

https://doi.org/10.1017/S0890060407000297 Published online by Cambridge University Press

interaction. However, interactions are primarily goal-oriented as they aim to produce behavior in the agents to fulfill specific functions. For example, an architect usually starts interacting with a costing expert to make that expert deliver cost information of a building, which is a behavior that implies the function of preparing cost evaluation. As a condition for realizing this behavior, the architect has to provide the costing expert with relevant input data that adds to and is consistent with that expert's knowledge structure. Functions comprise not only formally assigned tasks but also informal roles, such as spreading enthusiasm, providing critical feedback, and mediating between different viewpoints. Integrating a good mix of informal functions in a team based on distinct personality types is known to stimulate team dynamics and productivity (Bradley & Hebert, 1997).

Interactions between agents consist of sequences of social actions performed by individual agents. Social action has been defined as a type of action whose "subjective meaning takes account of the behavior of others and is thereby oriented in its course" (Weber, 1968). Adding to Weber's definition, we can describe social actions as both purposeful and constrained by the cognitive and physical capabilities of the agent to which the action is directed. This characterization of social action shares the basic traits of designing. Indeed, we can view social action as an activity that "designs" parts of another agent's knowledge structure to produce a behavior reaction that serves some function.

Let us apply the situated FBS framework to social action, using the example of the architect and the costing expert. The internal world refers to the architect.

1. *Formulation:* The architect's current goal of checking if a candidate solution for a building design meets given budget constraints drives the construction of the functional requirement "to prepare cost evaluation." This function is specified either by an external agent such as the architect's supervisor (interpreted via process 1 labeled in Fig. 4) or by the architect themselves (process 4). Additional requirements are constructed that relate to appropriate behavior such as the type of information to be produced or time constraints for information delivery. These requirements may be produced externally (process 2) or internally (process 5). Structure requirements include the class of expertise or the expert necessary to produce the required behavior, stated explicitly (process 3) or constructed internally (process 6). In our example, this formulated structure may include knowledge about the amounts and kinds of building materials used or the land area occupied by the building. Formulation concludes after transferring the requirements into the expected world (processes 7, 8, and 9) and eventually deriving additional behaviors (process 10).
2. *Synthesis:* A structure (Se^i) is generated (process 11) that instantiates the relevant pieces of knowledge required by the costing expert to exhibit the expected behavior. These pieces of knowledge have to be consistent with the preexisting knowledge structure of the costing expert for reasons of interoperability. This includes that expert's terminologies and representation formats. The externalization of the expected structure (via process 12) corresponds to the architect's communicative action directed to the costing expert.
3. *Analysis:* This process is performed by the costing expert, who produces interpreted structure (S^i, process 13) and derives a behavior reaction (process 14).
4. *Evaluation:* the architect compares the costing expert's actual behavior (B^i) with the expected behavior (Be^i, process 15), to determine if that expert succeeded or failed to deliver the required type of information.
5. *Documentation:* In this context, documentation represents a form of "metacommunication" about the costing expert to some third party. An example is the architect chatting with a colleague about the interaction with the costing expert, in terms of the architect's goals (function) and the costing expert's responses (behavior) and/or assumed beliefs and goals (structure).
6. *Reformulation type 1:* Reformulation of structure (Se^i) via focusing (process 9) may be needed in case of unsatisfactory evaluation. For example, the costing expert might have failed to produce a result because of incomplete knowledge about the building design data relevant for performing the cost analysis. Inferring the specific reason for such failure and the means to address that reason can be performed internally by the architect (process 6) or with the support of external representations of the costing expert's knowledge provided by the costing expert (process 13) or required by the architect's supervisor (process 3).
7. *Reformulation type 2:* Reformulation of behavior (Be^i) via focusing (process 8) may occur by relaxing constraints on behavior such as the time required for getting the results from the costing expert. One possible driver for this process is the derivation of behavior from structure (process 14). For example, the architect may detect difficulties (such as redundancies or formatting problems) in the initial formulation of input data given to the costing expert, which to resolve requires substantial amounts of time. Another potential driver is the interpretation of explicitly represented behavior, such as reported times of previous costing tasks (process 19) or a notification from the supervisor that longer delays could be accepted (process 2). Finally, constructive memory (process 5) may drive behavior reformulation. An example is the architect's growing experience (possibly gained through being involved in other design projects run in parallel) leading to changed expectations about the time needed for cost analysis.
8. *Reformulation type 3:* reformulation of function (Fe^i) via focusing (process 7) may take place as a result of changes in the domain tasks to be carried out. For example, major changes in the product requirements (e.g., the

https://doi.org/10.1017/S0890060407000297 Published online by Cambridge University Press

client may have requested an important feature to be added to the building design) may postpone the cost evaluation that produced the functional requirement for interacting with the costing expert. Other functions now become relevant based on different, upstream tasks to be carried out, such as conceptual design and structural analyses. These new functions may originate from the architect's external world, in the form of potential functions (process 20) or explicitly requested functions (process 1), or from the architect's internal world (process 4). Another driver for function reformulation may be the ascription of a new function to behavior (process 16). For example, the costing expert, who is a former architecture student, may point out a minor flaw in the building design. This may add the function "to support design verification" to the initial function "to prepare cost evaluation."

The view of social action as an instance of designing is consistent with Gero and Kannengiesser's (2004*b*) model of common ground. The FBS model of the costing expert (as the "design object") is constructed not exclusively by the architect (as the "designer") but in collaboration with the costing expert. This is a consequence of the necessity in designing to align the expected world with the external world. Although all FBS representations in the expected world can be autonomously controlled by the "designer," the FBS representations in the external world mainly depend on how the other agent (the "design object") represents itself. For the "design" to be successful, both agents must agree on a common FBS representation, which may involve adjustments in their expected FBS models of each other and social actions in both directions. The result of agents forming common ground is often referred to as mutual trust, which has been recognized as important in building successful teams (Kramer & Tyler, 1996).

4.2. Emerging team structures

The FBS ontology provides a uniform schema for structuring and generalizing experiences with a variety of objects, agents, processes, and so forth. These are activities that are vital to understand and predict states of affairs and courses of events in a complex, dynamic world. A structured, generalized way of internal representation leads to a certain amount of continuity both in the actions performed by the agent and the results or reactions produced in the environment. The perceived patterns of interactions provide the grounds for further generalized constructs, namely the notion of relationships. Recurrent patterns of social interaction between agents are accordingly generalized as social relationships or coordination structures (Malone, 1987).

Some social relationships in design teams are predefined and used to compose formal hierarchical or network structures. Other social relationships can emerge independently from formal ones, that is, without or in addition to team structures that have been explicitly specified. We will refer to these relationships as informal. Consider the example of a large design team involving a consortium of several companies from different countries. Here, a group of engineers of an English company A is to coordinate their design with engineers of a Chinese company B (Fig. 7a). As one of the engineers of A turns out to have a certain amount of knowledge about Chinese language and culture, he is allocated by his colleagues (with his consent) the function "to provide a liaison with the Chinese partner." As a result, a new set of interactions commence between the new "liaison engineer" and his colleagues, namely those concerned with providing that engineer with all design information that is relevant for coordination with the engineers of company B. These interactions lead to the establishment of informal relationships within company A's engineering group (Fig. 7b). Likewise, a set of informal cross-company relationships forms between the liaison engineer and some of company B's engineers, established through their regular interactions.

New team structures may not only emerge as a set of additional, informal relationships, but also as a modified set of team components. New team components may be formed by integrating new agents into the team. This process is based on the same principle as illustrated in the previous example, namely on the use of FBS models of individual agents. Here, FBS models inform interactions that traverse the

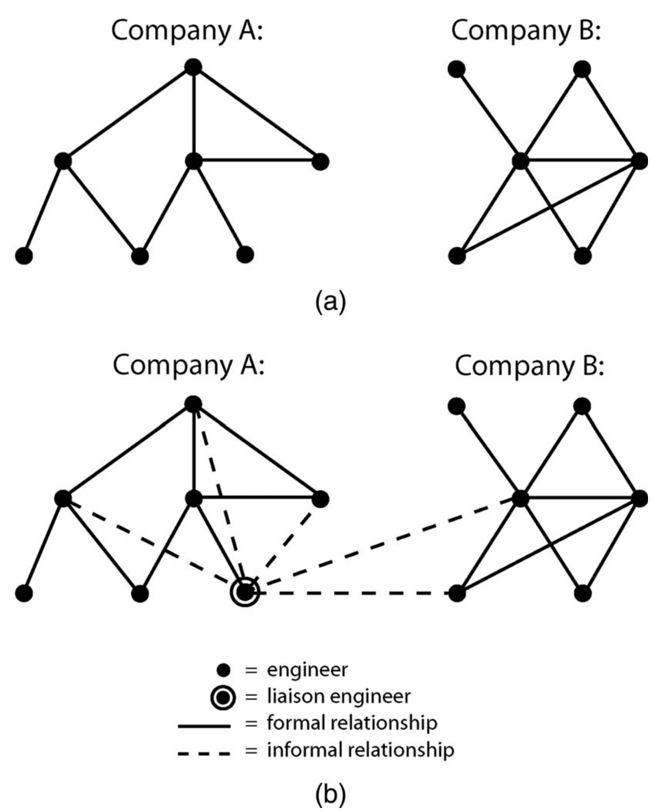


Fig. 7. Relationships among members of two companies: (a) formal relationships established at the outset of the design project and (b) informal relationships emerge as a result of interactions with the liaison engineer.

boundaries of the existing team, eventually leading to new relationships with external agents as if they were part of the team. For example, a team of junior engineers, faced with a difficult design problem, may seek expert advice from a senior engineer who is not a member of the team. The senior engineer's input to the team's problem-solving process and his interest in the design may be so substantial that he becomes more regularly involved in the team's activities. Over time, the boundary of the team becomes somewhat blurred as a new informal relationship forms with the nonmember.

Adding new components to a team does not necessarily involve integrating external agents. Components may also be built from existing team members, resulting in more complex entities within the team. These entities are often referred to as communities of practice (CoPs; Wenger, 1998). Lave and Wenger (1991) have defined a CoP as "a set of relations among persons, activity and world, over time and in relation with other tangential and overlapping communities of practice." CoPs bring together practitioners of a domain engaging in a common enterprise and creating shared histories of interacting and learning that differentiate the participants of a CoP from nonparticipants (Wenger, 1998; Fischer, 2001). CoPs are entities in their own right, exhibiting behaviors that cannot be easily explained by looking at the individual level alone. CoPs develop their own identities, conventions, and standards, which strongly influence individual choices, such as the evaluation and adoption of different design solutions (Sosa & Gero, 2005).

CoPs correspond to what may be called informal teams. These informal teams are often located within larger, formal teams, in which case they form subteams. Multiple subteams may exist within a team, possibly including various overlaps. Figure 8 illustrates this for a set of six agents, all of which are members of the same formal team. Within this team there are four subteams: 1 ∩ 2, 1 ∩ 2 ∩ 3, 3 ∩ 4, and 5 ∩ 6.

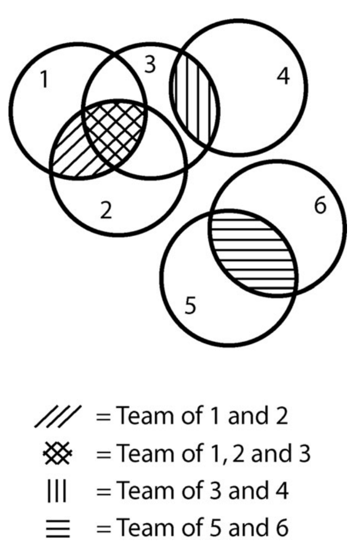


Fig. 8. Subteams and relationships within a set of six agents (1, 2, 3, 4, 5, and 6). Circles represent agents, intersections represent relationships, and hatched intersections represent those relationships that compose a team.

Not every relationship among a set of agents leads to a view of them as an informal team. This concerns groups of interrelated agents that are not engaged in activities considered useful from a systems perspective. For example, although a group of colleagues, by meeting every day to have lunch together, is engaged in a common activity that might serve private, recreational purposes, this activity lacks the aspect of usefulness for the overall team or organization. Unless this group uses their lunch break for developing new ideas for product or process improvement for the global or a superordinate team, there are no grounds for regarding that group as an informal team.

The FBS schema is useful for distinguishing between informal teams and other groups of agents, as it adds function and behavior descriptions to structure features of a team or group of agents. Team structures exhibit behaviors that serve functions in accordance with superordinate goals. Other group (nonteam) structures represent behaviors that fulfill either no functions at all or no functions associated with any higher level goal.

The capacity of the FBS schema to represent all agents and all teams, formal and informal, can be used to reason about and evaluate different t-design alternatives. Agents or teams that do not perform well or that are of poor use in the current functional context may be replaced by other agents or teams with different structures, behaviors, or functions. Eventually, emerging informal teams or informal relationships may be formalized to substitute or coexist with predefined formal team structures.

FBS representations of informal teams may also be used for social interaction of an individual agent with a team or for social interaction between two teams. The common ground needed for the latter is established by the teams constructing consistent FBS models of each other. This idea may be illustrated by a simple adaptation of Figure 5, replacing all occurrences of the term "agent" by "team." Here, the structure part of the FBS models may be interpreted as referring to the team's composition and to the team's common ground (as a generalized representation of the team's collective knowledge).

The processes involved in social actions between different teams can all be represented in the FBS framework, in a similar way as outlined for individual agents in Section 4.1. This allows modeling interactions within heterogenous design teams, so-called communities of interest (Fischer, 2001), consisting of CoPs from different disciplines. The FBS schema provides communities of interest with a set of ontological categories to relate the communicative behavior of a CoP both with the current functional context and constraining cognitive and social structures within a CoP. This makes interactions adapted to the goals (captured as functions) and the capabilities (captured as structure) of different CoPs.

The ability to represent all CoPs uniformly in terms of FBS allows constructing generalizations that may compensate for missing specific information about a CoP. This effect is the same as for interactions between individual agents. We can adapt Figure 6 to represent this concept by replacing all occurrences of the term "agent" by "team." Here, a team 0 (or a representative member of team 0) has constructed FBS models of four other teams (teams 1, 2, 3, and 4), with teams 1 and 2 being better known than teams 3 and 4. For interactions with team 4, the FBS model of that team is augmented by deriving generalized knowledge from FBS models of similar teams. Generalizations may also provide sufficient information for interacting with completely new teams such as team 5.

Interactions with a team can be represented by the same fundamental processes as for representing interactions between individual agents, defined in the situated FBS framework. This framework can capture interactions that occur between different teams, interactions that occur between a team and an individual representative of another team and interactions that occur between two individual representatives of different teams.

The FBS schema supports the ability of individual agents and teams to reason about all entities at all levels of aggregation, from the overall team level to the individual agent level. On the one hand, this provides system stability by propagating global team properties down to the local components. Agents are inclined to adhere to these structures when engaging in local interactions. In contrast, there is a certain degree of flexibility induced by separating task hierarchies from organizational hierarchies (Mesarović et al., 1970). Our approach represents task hierarchies as the functions and organizational hierarchies as the structure of teams. Connections between functions and structures at all levels can be indirectly established by every agent or team, validated via comparing relevant behaviors and reformulated individually or collaboratively.

5. CONCLUSION

We have presented an ontology of situated design teams, based on an existing framework of situated designing. Specifically, we have derived this ontology from two new applications of the situated FBS framework in regard to the following:

1. *The subject of designing:* By generalizing the interactions between the expected, interpreted and external worlds, we have extended the original focus of the situated FBS framework to include the notion of situated designing carried out by teams rather than just by individual designers. This provides an ontological framework for representing, analyzing, and understanding the effects of situated cognition at a team level on the product that is designed.
2. *The object of designing:* By viewing teams in terms of function, behavior, and structure, we have presented three ontological levels at which changes in design teams can be studied. The situated FBS framework represents these changes as the outcomes of a purposeful (meta-)design activity, driven by a set of distinct processes involving situated interaction.

The second of these two applications is related to work in organizational self-design (Corkill & Lesser, 1983) and virtual design teams (Levitt et al., 1994), and can be viewed as an instance of configuration design (Brown, 1998). Our approach differs from this work through our focus on the situatedness of team designing. We have followed the idea of a team as the object of designing in a consistent way: from the overall team level to the individual agent level. In particular, we have modeled social interaction as a form of situated designing. This opens up new ways of understanding common ground in multiagent systems.

Applying the situated FBS framework at all levels of aggregation allows describing changes in a team, induced by situated interaction, at the needed level of detail. This provides a formal basis for modeling, understanding, and analyzing emergent structures in design teams.

ACKNOWLEDGMENTS

This research, which was conducted while the authors were in residence at the Key Centre of Design Computing and Cognition, University of Sydney, is supported by a grant from the Australian Research Council (Situated Design Computing Grant DP0559885).

REFERENCES

Bartlett, F.C. (1932, reprinted in 1977). *Remembering: A Study in Experimental and Social Psychology*. Cambridge: Cambridge University Press.
Bickhard, M.H., & Campbell, R.L. (1996). Topologies of learning. *New Ideas in Psychology 14(2)*, 111–156.
Bradley, J.H., & Hebert, F.J. (1997). The effect of personality type on team performance. *Journal of Management Development 16(5)*, 337–353.
Brown, D.C. (1998). Defining configuring. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing 12(4)*, 301–305.
Clancey, W.J. (1997). *Situated Cognition: On Human Knowledge and Computer Representations*. Cambridge: Cambridge University Press.
Clark, H.H. (1996). *Using Language*. Cambridge: Cambridge University Press.
Cohen, P.R., & Levesque, H.J. (1991). Teamwork. *Noûs 25(4)*, 487–512.
Corkill, D.D., & Lesser, V.R. (1983). The use of meta-level control for coordination in a distributed problem-solving network. In *Int. Joint Conf. Artificial Intelligence ‘83* Bond, A.H., & Gasser, L., Eds.), pp. 748–755, Karlsruhe, Germany.
de Kleer, J., & Brown, J.S. (1984). A qualitative physics based on confluences. *Artificial Intelligence 24*, 7–83.
Dewey, J. (1981). The reflex arc concept in psychology. *Psychological Review 3*, 357–370.
Fischer, G. (2001). External and sharable artifacts as opportunities for social creativity in communities of interest. In *Computational and Cognitive Models of Creative Design V* (Gero, J.S., & Maher, M.L., Eds.), pp. 67–89. Sydney, Australia: Key Centre of Design Computing and Cognition, University of Sydney.
Galbraith, J.R. (1977). *Organization Design: An Information Processing View*. Reading, MA: Addison–Wesley.
Gero, J.S. (1990). Design prototypes: a knowledge representation schema for design. *AI Magazine 11(4)*, 26–36.
Gero, J.S. (1999). Constructive memory in design thinking. In *Design Thinking Research Symp.: Design Representation* (Goldschmidt, G., & Porter, W., Eds.), pp. 29–35. Cambridge, MA: MIT.
Gero, J.S., & Fujii, H. (2000). A computational framework for concept formation for a situated design agent. *Knowledge-Based Systems 13(6)*, 361–368.
Gero, J.S., & Kannengiesser, U. (2004*a*). The situated function–behaviour–structure framework. *Design Studies 25(4)*, 373–391.
Gero, J.S., & Kannengiesser, U. (2004*b*). Modelling expertise of temporary design teams. *Journal of Design Research 4(2)*.

Kannengiesser, U., & Gero, J.S. (2007). Agent-based interoperability without product model standards. *Computer-Aided Civil and Infrastructure Engineering 22(2)*, 80–97.

Kramer, R.M., & Tyler, T.R., Eds. (1996). *Trust in Organizations: Frontiers of Theory and Research*. Thousand Oaks, CA: Sage.

Lave, J., & Wenger, E. (1991). *Situated Learning: Legitimate peripheral participation*. Cambridge: Cambridge University Press.

Levitt, R.E., Cohen, G.P., Kunz, J.C., Nass, C.I., Christiansen, T.R., & Jin, Y. (1994). The virtual design team: simulating how organization structure and information processing tools affect team performance. In *Computational Organization Theory* (Carley, K.M., & Prietula, M.J., Eds.), pp. 1–18. Hillsdale, NJ: Erlbaum.

Malone, T.W. (1987). Modeling coordination in organizations and markets. *Management Science 33(10)*, 1317–1332.

Mesarović, M.D., Macko, D., & Takahara, Y. (1970). *Theory of Hierarchical, Multilevel, Systems*. New York: Academic Press.

Minsky, M. (1985). *The Society of Mind*. New York: Simon & Schuster.

Nonaka, I. (1994). A dynamic theory of organizational knowledge creation. *Organization Science 5(1)*, 14–37.

Schön, D.A., & Wiggins, G. (1992). Kinds of seeing and their functions in designing. *Design Studies 13(2)*, 135–156.

Smith, G.J., & Gero, J.S. (2005). What does an artificial design agent mean by being "situated"? *Design Studies 26(5)*, 535–561.

Sosa, R., & Gero, J.S. (2005). A computational study of creativity in design: The role of society. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing 19(4)*, 229–244.

Suwa, M., Gero, J.S., & Purcell, T. (1999). Unexpected discoveries and s-inventions of design requirements: a key to creative designs. In *Computational Models of Creative Design IV* (Gero, J.S., & Maher, M.L., Eds.), pp. 297–320. Sydney, Australia: Key Centre of Design Computing and Cognition, University of Sydney.

Weber, M. (1968). *Economy and Society: An Outline of Interpretive Sociology*. New York: Bedminster Press.

Wegner, D.M. (1986). Transactive memory: a contemporary analysis of the group mind. In *Theories of Group Behavior* (Mullen, B., & Goethals, G.R., Eds.), pp. 185–208. New York: Springer–Verlag.

Wenger, E. (1998). *Communities of Practice: Learning, Meaning and Identity*. Cambridge: Cambridge University Press.

Ziemke, T. (1999). Rethinking grounding. In *Understanding Representation in the Cognitive Sciences: Does Representation Need Reality?* (Riegler, A., Peschl, M., & von Stein, A., Eds.), pp. 177–190. New York: Plenum Press.

John S. Gero is a Research Professor at the Krasnow Institute for Advanced Study and the Department of Computer Science, George Mason University, and Visiting Professor at the Massachusetts Institute of Technology. He was formerly a Professor of design science and Co-Director of the Key Centre of Design Computing at the University of Sydney. He is the author or editor of 43 books and has published over 550 research papers. He has been a Visiting Professor of architecture, civil engineering, cognitive psychology, computer science, design and computation, and mechanical engineering in the United States, United Kingdom, France, and Switzerland. His research focuses on computational, cognitive, and neurocognitive studies of designing.

Udo Kannengiesser is a Researcher at NICTA, Australia's Centre of Excellence for information and communication technology (ICT). He obtained his PhD at the Key Centre of Design Computing and Cognition at the University of Sydney. He holds a degree in mechanical engineering from the University of Karlsruhe, Germany, with a specialization in information systems for design and production.