European Journal of Information Systems (2006) 15, 91–102

www.palgrave-journals.com/ejis

Process models representing knowledge for action: a revised quality framework

John Krogstie[1,2], Guttorm Sindre[1] and Håvard Jørgensen[3]

[1]Norwegian University of Science and Technology, Trondheim, Norway; [2]SINTEF ICT, Oslo, Norway; [3]Troux Technologies, Oslo, Norway

Correspondence:
John Krogstie, Norwegian University of Science and Technology, Sem Sælands vei 7-9, Trondheim NO-7034, Norway.
Tel: +47 93417551;
E-mail: krogstie@idi.ntnu.no

Abstract
A semiotic framework for evaluating the quality of conceptual models was proposed by (Lindland OI, Sindre G and Sølvberg A (1994) Understanding Quality in Conceptual Modelling, IEEE Software **11(2)**, 41–49) and has later been extended in several works. While the extensions have fixed some of the limitations of the initial framework, other limitations remain. In particular, the framework is too static in its view upon semantic quality, mainly considering models, not modelling activities, and comparing these models to a static domain rather than seeing the model as a facilitator for changing the domain. Also, the framework's definition of pragmatic quality is quite narrow, focusing on understanding, in line with the semiotics of Morris, while newer research in linguistics and semiotics has focused beyond mere understanding, on how the model is used and impact its interpreters. The need for a more dynamic view in the semiotic quality framework is particularly evident when considering process models, which themselves often prescribe or even enact actions in the problem domain, hence a change to the model may also change the problem domain directly. This paper discusses the quality framework in relation to active process models and suggests a revised framework based on this.
European Journal of Information Systems (2006) **15,** 91–102.
doi:10.1057/palgrave.ejis.3000598

Keywords: action and process modelling; interactive models; knowledge creation; model quality

Introduction

Models are generally defined as explicit representations of some portions of reality as perceived by some actor (Wegner & Goldin, 1999). Modelling in various forms is essential in supporting complex human design activities. In the development of information systems, as well as the re-engineering of work practices, the modelling of business processes or workflows often plays a central part (Hjalmarsson & Lind, 2004). A model is *active* if it directly influences the reality it reflects, that is, changes to the model also change the way some actors perceive reality. Actors in this context include users as well as software components. Greenwood *et al.* (1995) argue that active models can enable information systems to meet many business needs that current technologies fail to support.

Model activation is the process by which a model affects reality. Model activation involves actors interpreting the model and to some extent adjusting their behaviour accordingly. This process can be

- *automated*, where a software component interprets the model,
- *manual*, where the model guides the actions of human actors, or
- *interactive*, where prescribed aspects of the model are automatically interpreted and ambiguous parts are left to the users to resolve.

Received: 17 May 2005
Revised: 16 December 2005
Accepted: 3 January 2006

Fully automated activation implies that the model must be formal and complete, while manual and interactive activation also can handle incomplete and partly informal models. Completing this terminology, we define a model to be *interactive* if it is interactively activated. That a model is interactive entails a co-evolution of the model and its domain. A model that does not change will not be able to reflect aspects of reality that changes, nor can it reflect evolution of a human actor's understanding. Consequently, an interactive model that does not evolve will deteriorate. It contributes to change but does not reflect this change. The process of updating an interactive model is called *articulation*. The interplay of articulation and activation reflects the mutual constitution of interactive models and the social reality they reflect. The software components that support intertwined articulation and activation are termed *model activators*.

The most comprehensive theoretical approach to this field is Peter Wegner's interaction framework (Wegner, 1997; Wegner & Goldin, 1999). Its development was triggered by the realisation that machines involving users in their problem solving could solve a larger class of problems than algorithmic systems computing in isolation (Wegner, 1997). The primary characteristic of an *interaction machine* is that it can pose questions to users during its computation. The process can be a multi-step conversation between the user and the machine, each being able to take the initiative. The notion of an interaction machine is further extended to that of *multi-stream distributed interaction machines*, enabling multiple users and external systems to interact simultaneously, for example, in groupware systems. Workflow (Carlsen *et al*., 1997), document classification and retrieval (Brasethvik & Gulla, 1999), product data management (Farshchian, 2000), co-operation support, knowledge-based and reflective systems are areas where active and interactive models have been applied. DEUDU (Bøving & Petersen, 2002) is a concept for system adaptability by user intervention, as a contributing designer, at use time.

In much knowledge management literature, inspired by (Nonaka, 1991), a model of a business process, whether active or not, would be considered externalised knowledge about the organisation. However, this 'knowledge-as-object' view has been criticised by Walsham (2004, 2005), maintaining that Nonaka misinterpreted the distinction between explicit and tacit knowledge (Polanyi, 1966) and that *knowledge* is something within the human mind, so that the term should not be used for passive representations of information in writing or in computer systems. Other researchers are in line with this view, furthermore stressing that an essential quality of knowledge lies in its ability to support action (Braf, 2004). Hence, knowledge is not only about action, but also *for* action, or even *part of* action (Cook & Brown, 1999). The same view pertains to information systems, which can be seen as (partial) automations of conceptual models of the problem domain. Information Systems Actability Theory (Goldkuhl & Ågerfalk, 2002; Ågerfalk, 2003) stresses that information systems are action systems used in a social action context (Ågerfalk & Eriksson, 2003).

We support the criticism of the 'knowledge-as-object' view; hence, in this paper a model (e.g., of a business process) is not as such considered to *be* knowledge, but it may *contribute to knowledge* when interpreted (and acted upon) by a human or other intelligent agent. Since process models describe – or even prescribe – specific paths of action under specific circumstances, the road from interpretation of the model to action may be very short, especially for interactive models where changes have immediate effect on system behaviour. This also makes it especially important to understand and be able to evaluate the quality of such models.

As conceptual models can be considered as sets of statements in a language, it may be interesting to evaluate them in semiotic/linguistic terms. A semiotic framework (henceforth called the SEQUAL framework) for evaluating conceptual models was originally proposed in Lindland *et al*. (1994). In its initial version, it considered three quality levels: syntactic, semantic, and pragmatic quality. The framework was later extended (Krogstie *et al*., 1995; Krogstie & Jørgensen, 2002), adding more levels of Stamper's semiotic ladder (Stamper, 1996).

This paper will arrive at a revised SEQUAL framework to make it more appropriate for (active) process models. The remainder of the paper is structured as follows: in the next section, we review the previous framework (i.e., before revision). Then, we discuss the particular needs of process models (and especially active models), which is the targeted emphasis for the improvement of the framework. The following section presents a revised framework, before providing conclusions and ideas for further work.

The previous SEQUAL framework and its shortcomings

The previous full version of the framework can be seen in Figure 1. We have taken a set-theoretic approach to the discussion of model quality at different semiotic levels, which has been defined as the correspondence between statements belonging to the following sets:

- G, the (normally organisationally defined) goals of the modelling.
- L, the language extension, that is, the set of all statements that are possible to make according to the graphemes, vocabulary, and syntax of the modelling languages used.
- D, the domain, that is, the set of all statements that can be stated about the situation at hand. Note that the domains we are normally dealing with in process modelling have been socially constructed, and are more or less inter-subjectively agreed. The view that the world is socially constructed does not make it any less important to model that world, to be able to

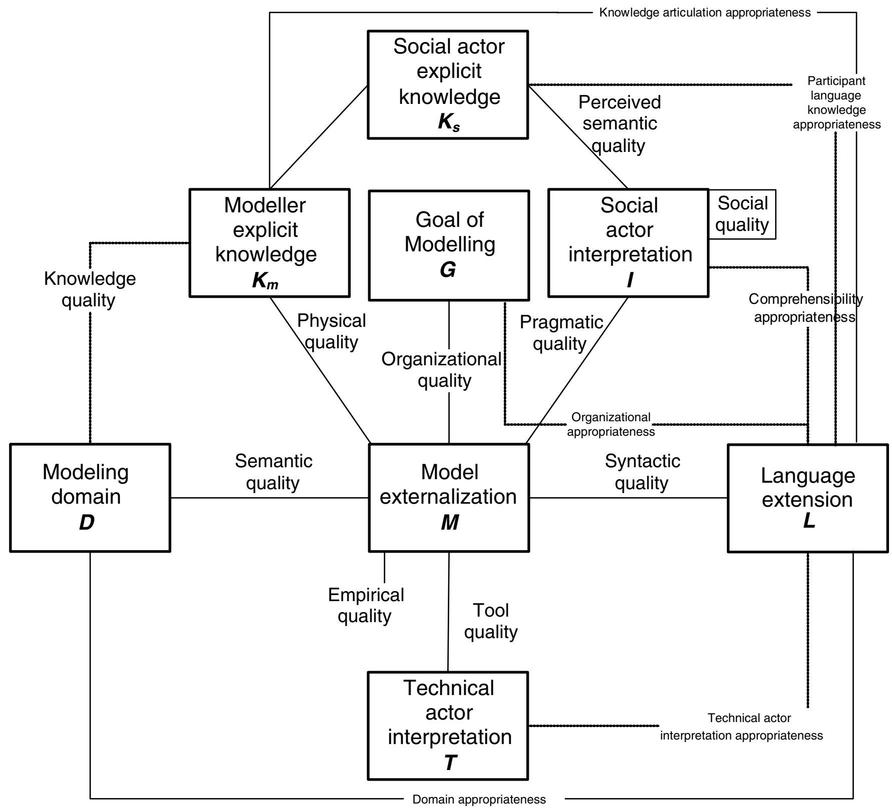

Figure 1 SEQUAL model quality framework, previous version.

understand and potentially reconstruct it (Dahlbom, 1991).

- *M*, the externalised model, that is, the set of all statements in someone's model of part of the perceived reality written in a language.
- K_s, the relevant explicit knowledge of the set of stakeholders being involved in modelling (the audience *A*). A subset of the audience is those actively involved in modelling, and their knowledge is indicated by K_M.
- *I*, the social actor interpretation, that is, the set of all statements that the audience think that an externalised model consists of.
- *T*, the technical actor interpretation, that is, the statements in the model as 'interpreted' by the different modelling tools.

The framework separates quality goals from quality means. Quality is defined on different levels following the levels of the semiotic ladder (Stamper, 1996):

Physical quality has two goals: *externalisation* and *internalisability*. Externalisation means a model is available as a physical artefact, representing the knowledge of some social actor using statements of the modelling language. Internalisability means the model is available and persistently enabling the model audience to interpret it.

Empirical quality, as Figure 1 suggests, can be evaluated looking only at the model itself, comprising comprehensibility matters such as layout for graphs and readability indexes for text.

Syntactic quality has the goal of *syntactic correctness*, that is, that all statements in the model are according to the syntax and vocabulary of the modelling language.

Semantic quality has the quality goals of *validity* and *completeness*. Validity here means that all statements in the model are correct and relevant to the problem, while completeness means that the model contains all statements that *would be* correct. Since we cannot proceed with modelling endlessly, we have introduced the notion of *feasibility* relative to the goals of modelling (relative to organisational quality as discussed below); that is, we relax upon validity and completeness by letting the modelling process end when the model has reached a state where further modelling is regarded less beneficial than that by accepting the model in its current state. Consistency checking is regarded as a mean to help achieve feasible validity and feasible completeness. Other semantic quality means are formal semantics, model reuse, and modifiability of the models.

Perceived semantic quality covers the correspondence between actors' interpretation of the model and their current knowledge of the domain and has the goals of *perceived validity* and *perceived completeness*. Important means to achieve high perceived semantic quality are the same as those for achieving semantic quality, with the addition of participant training and *requisite variety*.

Pragmatic quality is the correspondence between the model and the audience's interpretation of the model and has one goal, *comprehension*, meaning that the model has been understood. Means to increase pragmatic quality include not only executability, animation, and simulation but also more advanced techniques like model transformations, model filtering to present model abstractions from several viewpoints, model translation, and explanation generation.

Social quality has the goal of *feasible agreement* where agreement covers agreement in knowledge, agreement in interpretation, and both relative and absolute agreement. Feasible agreement does not have to imply consensus, it only implies resolving inconsistencies by choosing alternatives where benefits of choosing exceed the costs of working out consensus. Means to achieve high social quality include inconsistency handling, model integration, and conflict resolution, including the so far seemingly untapped potential of linking modelling tools with argumentation support tools.

Organisational quality sees that the model is fulfilling the goals of modelling in the first place.

Shortcomings: Some empirical testing of using the framework for evaluating models was carried out in Moody *et al*. (2002a and 2003), showing that the participants had problems in making reliable evaluations by means of it. While the set algebra definitions of various quality goals provide some clarity as to what the different quality levels mean, it is really only syntactic quality (of the levels in the original framework), which has any hope of being objectively measured, as both the problem domain and the minds of the stakeholders are unavailable for formal inspection. This would, however, be a general challenge for any framework trying to address model evaluation beyond the syntactic level.

Some other shortcomings are more directly relevant for active models:

(1) The framework takes a quite static view of the domain *D*. The domain somehow just *is* there, and the quality of the model is considered relative to it, as well as of the stakeholders' knowledge *K*. This suggests that the initial version of the framework was conceived with a descriptive kind of conceptual modelling in mind (i.e., 'as-is' modelling to understand the current situation in the early stages of problem analysis in an information systems project). But process models are often prescriptive rather than descriptive, and this certainly applies to active models. Then, a comparison with an existing problem domain to establish semantic quality becomes awkward. Hence, semantic quality must be defined differently in the revised version.
(2) Pragmatic quality is quite narrowly defined in the SEQUAL framework (in the tradition of Morris) – is the model understood? – while in newer works within linguistics and semiotics, pragmatic quality is normally a much wider notion concerning the effects of a message (Goldkuhl, 2002). This more modern view fits better for interactive models where the effect of a model change can be quite immediate. Hence, pragmatic quality must also be defined differently in the revised version.
(3) The framework's definitions can be understood as leaning towards the 'knowledge-as-object' view, for example, talking about the model as 'externalised knowledge'. In accordance with the criticism of this view, given in the Introduction, 'knowledge-as-object' phrases will be avoided in the definition of the revised framework.

Quality issues of (interactive) process models

An interactive model is immersed in its usage context. Model articulation and activation take place concurrently, and are mutually dependent upon each other. Agreement among actors is vital for information systems development, because it is costly to fix errors late in the project. For an interactive model, the costs of fixing errors are diminished. The goal of social quality for interactive models thus becomes to support social learning and construction of shared understanding.

Similarly, semantic and syntactic correctness becomes less important because users can be put in charge of resolving inconsistencies during interactive activation. For learning, the ability to represent inconsistent points of view is crucial. A system should thus not deny articulation of syntactically incomplete model fragments, but instead capture inconsistent views so that they can be negotiated.

Interactive activation implies that the goal of semantic completeness is replaced by a goal of letting the users articulate their reality at the level of detail and specificity that they find useful. Incompleteness can be resolved at the time of activation.

More fundamentally, the existing model quality framework embodies a static, structural perspective. It discusses relationships between sets of statements about the world. Interactive modelling quality (Krogstie & Jørgensen, 2002) demands a more dynamic approach focusing on how these sets of statements interact and influence each other in the interdependent processes of articulation, activation and reuse. This perspective is conceptualised below, defining core processes as sequences of activities that transform statement sets in the quality framework model. A focus on core processes rather than static quality parameters emphasise interdependencies and trade-offs, which designers of interactive systems must resolve.

Figure 2 taken from Jørgensen (2004) summarises this perspective. The rest of this section elucidates basic concepts in this framework. Below, italics refer to associations in Figure 2.

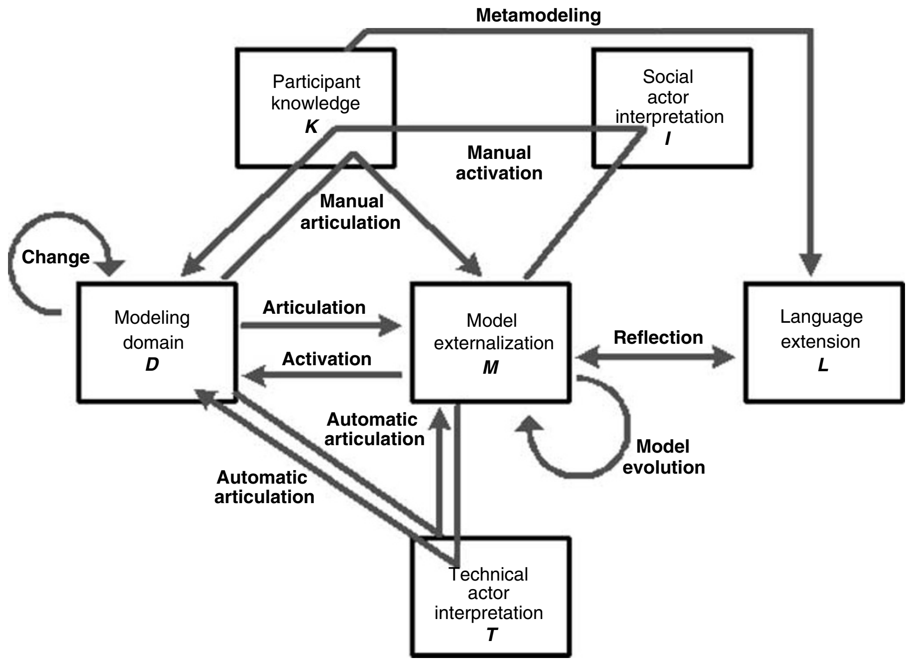

Figure 2 Interactive processes in the model quality framework.

Articulation (D→M): Articulation is the process where domain features are represented in the model (*M*) by means of the modelling language (*L*). It should (but does need to) increase the semantic quality of the model.

Manual articulation (D→K→M): Most articulation is manual, involving the external representation of participant knowledge (*K*) about the domain (*D*) into the model (*M*) using the language (*L*). In addition to empirical, physical, and semantic quality, it depends on the quality of participant knowledge about the domain, and the appropriateness of the language with respect to the domain and participants' knowledge.

Automatic articulation (D→T→M): Sensors are computerised components that capture information from the model domain, through interfaces to external information systems or hardware devices.

External change (D→D): Changes in the domain (*D*) may become known to the participants of the project (*K*). Known changes may become articulated into the model *M*.

Model evolution (M→M): Model evolution may be triggered by change and reflected through articulation, or it is motivated by the need to cause future domain change through activation.

Activation (M→D): Model activation involves model-guided actions that transform the domain (*D*).

Automatic activation (M→T→D): Automatic activation implies that the model is interpreted and acted upon by a computerised component, in our terminology a model *activator*. The *executability appropriateness* of the modelling language reflects its automatic activation potential.

Manual activation (M→I→K→D): Human actions based on an interpretation (*I*) of the model (*M*) constitute a manual activation process. The involved actors may also have created the model themselves (being participants in the modelling). Actions involve changing the domain *D*, and should thus be reflected in the model *M*. While automatic activation is deducible from the model and thus already captured, manual actions include elements of human model interpretation, which should be captured in the model.

Interactive activation (M→I↔T→D): Interactive activation exists in different forms. In many workflow systems, it is the technical component that tells the users what to do (e.g., by putting tasks in their inboxes). In other words, the modelled sequence of tasks is automatically interpreted, but the tasks are performed manually (*M→T→I→D*). In graphical user interfaces, the opposite sequence (*I→T→D*) is more common. Here it is up to the user to select which operation the system should perform next. If an action is the result of interactive or automatic model activation, its effects should be automatically captured in the model. This scheme increases the semantic quality without extra work for the users, completing the cycle *M→I↔T→D→M*.

Meta-modelling (K→L): Meta-modelling involves changing the modelling language. It is typically carried out in order to improve the comprehensibility of the language for human actors, or to make it more suitable in the local domain. Meta-modelling faces similar problems as reuse, but may also benefit from the immediate domain availability, which characterises interactive models. Local language adaptations should increase the pragmatic and semantic qualities of the resulting models.

As we understand, compared to traditional models used during the early phases of systems development, interactive models are faced with a different set of requirements. We summarise these using the sets in the quality framework, before outlining how these requirements influence the discussion of quality at the different levels within the quality framework.

- *G*, the goals of modelling is more related to have an up to date, tailored model to support the individual users in their actual tasks, and is less directly linked to overall goals of an organisation. Note that this is obviously a balance since one should not be able to change parts of the model that it is defined as very important for the organisation to keep stable (e.g., rules for performing financial transactions). Secondarily, the model should support organisational learning.
- *L*, the language extension: The languages used need to be simple enough, and represented in such a way that a large number of people with a diversity of backgrounds are able to use them. The language might be updated as part of modelling though meta-modelling.
- *D*, the domain: More closely related to the task at hand, and not to a generic support of an abstract domain. It is also more obvious to see how the modelling quite instantly is able to have organisational effect and change the domain through articulation of new knowledge.
- *M*, the currently externalised model: as before.
- K_m and K_s: A wider range of people are actively involved in modelling, amplifying the social, psychological, and organisational aspects. More people (potentially all users) act as modellers and model interpreters, thus the difference between the set of modellers and general social actor at least in the initial articulation is smaller.
- *I*, the social actor interpretation need to be able to quickly update the interpretation of model changes carried out by one person, which also influences other people.
- *T*, the technical actor interpretation: the model as interpreted by information systems. The models are available at run-time, but might only be partly defined, that is, the tools used must be made with interaction possibilities in mind, and be able to respond quickly to changes.

The main quality types are specialised in the following way:

Physical quality: For active models, the modellers include not only software professionals but also end-users. Hence, stronger requirements to support physical quality are likely, both because end-users lack experience with conceptual modelling, and one will want to update the models more frequently owing to learning, and this needs to be supported. The possibility to rapidly update the model (and thus the system) is one of the main advantages with this approach. In systems engineering, new approaches like incremental development and extreme programming attempt to shorten these learning cycles, but they are still hampered with a long time span from learning of the end-user to model and system change compared to what can be achieved with interactive models. Also, users are likely to have more in-depth knowledge *K* of their domain *D* than that of software developers who have seldom taken part in the practice of the domain. Consequently, the potential for high semantic quality at the time of activation is greater. Hence, simplicity, adaptability and user-orientedness of the modelling language are even more crucial for active and interactive models than for their traditional counterparts.

Empirical quality: Owing to the local nature of use of the extensions of interactive models, the requirements to an aesthetic layout are less than for models acting to present a common picture or standard process in an organisation.

Syntactic quality: As a result of providing user-oriented models, interactive model approaches will have to enhance formal syntax means by providing more functionality in the area of error detection, prevention, and recovery, especially regarding allowing the continued enactment of completed, error-free parts.

Semantic quality: Actions often involve changing the domain *D*, and should thus be reflected in the model *M*. If an action is supported by an information system with active models, it can be automatically captured, increasing the semantic quality of the active model without extra work for the users.

Pragmatic quality: The core of active models is how models are *activated*. Activation implies *interpretation* of the model and corresponding *action* by either the social or the technical actors. Technical pragmatic quality demands complete models with an operational semantics, while the social pragmatic quality of the models and the cognitive economy of externalisation ($K \rightarrow M$) often demands more flexible, informal approaches. Interactive model interpretation enables models with user-controlled levels of formality, detail and preciseness, bridging the gap between theory and practice.

This immediate nature of active models, stemming from the interaction of the real-world domain and active model, can also enhance the social pragmatic quality of the models. When both the real world and the model that reflects it are available and adaptable for the users, the connections between them are easier to understand. Simulation and training methods can be developed that utilises this connection. Zuboff's study of industrial control rooms (Zuboff, 1988) shows great benefits for users that are able to work both with the conceptual tools of the computer and the physical environment of the factory.

Social quality: In systems development, agreement among participants about the requirements is crucial since they form the basis for a lot of detailed technical work that cannot easily be redone. Active models have a more immediate connection to the system and the environment it represents, so users have also access to the domain when negotiating a shared understanding. Social quality is thus perhaps not as important when assumptions readily can be tested immediately in the real world.

If an active model is to be reused in another setting, agreement on semantics is more important. Social quality of active models influences the processes of negotiating meaning and domesticating reusable model fragments into the local situation and work practice (Voss *et al*., 2000). In these processes, the ability to represent conflicting interpretations and make local modifications is just as important as the ability to represent agreement (the end result) in an unambiguous way. Also, since people learn through their work and use of the models, agreement is likely to be partial and temporary.

Organisational quality: When using active models, the models are more a work tool for the individual and group, utilising instance models. Thus, the primary goal for the model to fulfil is those of the user and group in their situated action. When reusing models that of some reason are not to be changed (e.g., because they decode a procedure that need to be carried out in a certain way to be legally correct), these organisational rules have to be enforced by the support system, thus it must be possible to differentiate which parts of the model that can be changed and those that can not. Tracking of model elements back to organisational goals or earlier base-lined models is traditional means for supporting the organisational quality.

Interactive models should be coupled to organisational learning. At the organisational level, process models are particular knowledge representations resembling organisational images (Argyris & Schön, 1973). An organisation's theory of action is embedded in a behavioural world, which shapes and constrains its theory-in-use; to achieve double-loop learning – that is, 'doing the right things' instead of 'doing the things right' – models must include links to intentional aspects (Carlsen & Gjersvik, 1997). The gap between real and modelled processes has been highlighted as a major inhibiting factor of process support systems and organisational learning (Argyris & Schön, 1973) alike. Thus, active models have a great potential for flexibly supporting knowledge management and process improvement. At the group level, organisational learning and *social reality construction* influence how we view the creation, update, and enactment of process fragments. Organisations are socially constructed through joint action by involved social actors. Here, interactive models as an articulation of work play the role of boundary objects for supporting perspective making and perspective taking (Boland & Tenkasi, 1995). By giving users control over the models, we empower them to externalise and share knowledge.

Modelling as an action impacts as we have seen above the different sets in the quality framework. We briefly describe these impacts here, summarising potential aspects of pragmatics seen in this light:

- Change in goal *G*: On the basis of modelling, you might realise that the overall goal for your process should be updated, that is, double-loop learning.
- Change in language *L*: It might be found that the chosen language is not appropriate for modelling, and has to be enhanced or replaced, that is, meta-modelling takes place.
- Change in domain *D*: For example, the model makes one realise problems with the existing process, thus leads to process improvement.
- Change in knowledge *K*: Learning by the participants in modelling.
- Change in interpretation *I*: Through validation tasks, one often changes the understanding of the model.
- Change in tool interpretation: For instance, by developing new or changing existing model activators.

The revised framework for quality of process models

For simplicity, this discussion will not include the lowest three levels of the SEQUAL framework (physical, empirical, and syntactic) as these are anyway not the most problematic, only mentioned here that we redefine, physical quality with a more narrow interpretation than before. In the old definition physical quality considered whether the model was externalised (or only in somebody's mind), and therefore mandated a comparison between the knowledge (*K*) and the model (*M*). The correspondence between *K* and *M*, however, is a quite ambitious matter, not really fitting such a low level as the physical one – and moreover this gave our old definition of physical quality a 'knowledge-as-object' flavour. Hence, we restrict the notion of model to what is somehow articulated and reduces the concept of physical quality from looking at the correspondence between *M* and *K* to looking only at *M* as such, and how the model is physically represented (e.g., electronically or on paper, single copy or duplicated, stored where?, etc.) and available to stakeholders. Empirical and syntactic quality are taken to mean the same as before and are thus not shown in the outline of the revised framework.

Similarly, we leave the discussion of the highest levels of the previous framework (social, organisational) for further work as we consider it most important to get a useful definition of semantic and pragmatic quality before moving on. Moreover, a successful redefinition of pragmatic quality in the framework might mean that higher levels become obsolete, see for example, the criticism raised against the semiotic ladder that the distinction between the pragmatic and social levels is unclear (Goldkuhl & Ågerfalk, 2002).

Figure 3 illustrates the basic ideas of the revised framework. The sets can be explained as follows:

M is the model, which may be changed by a modelling activity (oval to the right). *D* is the domain, which is

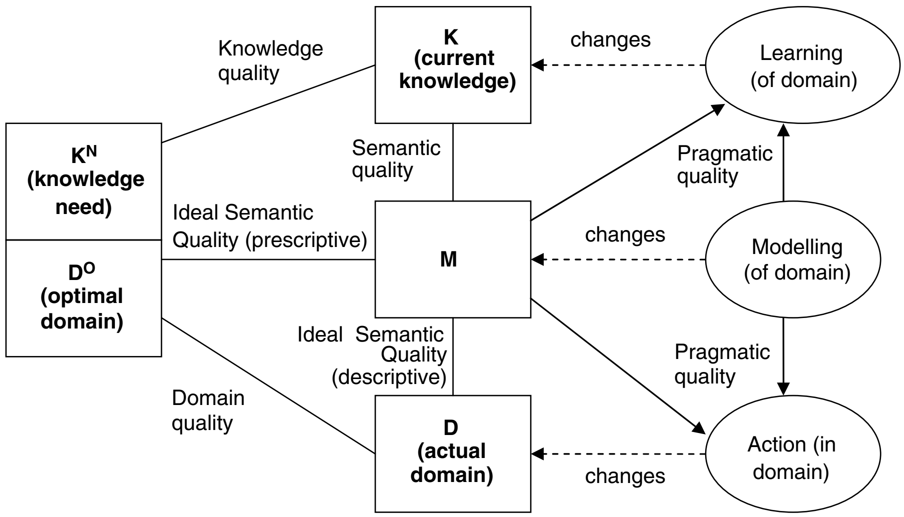

Figure 3 The revised framework.

similarly changed by actions taking place in the domain. This change may be facilitated by the model, or even directly caused by the modelling activity itself (e.g., in the case of active models). D^O is the optimal domain, that is, the situation that the organisation would have (or should have) wanted. This concept is useful for discussing whether a domain change is to the better or not. Of course, it is too simplistic to assume that the ideal domain will be the same for every member of an organisation; there will probably be as many different opinions about this as there are members in the organisation. On the other hand, there are areas that are decided outside the realm of the organisation (e.g., juridical laws and laws of nature).

K is the knowledge of the people in the organisation before the modelling activity, and this can change by learning, possibly facilitated by the model (e.g., employees acquiring new knowledge by looking at a model) or even caused by the modelling activity itself (e.g., employees learning something new by participating in a modelling workshop). K^N is the 'knowledge need', that is, the knowledge needed by the organisation to perform its tasks. In the previous version of SEQUAL, knowledge quality was seen as a relationship between *K* and *D*. K^N provides a more realistic point of comparison for *K* than for *D* since in many cases, it may not even be productive to know the full truth about the domain.

Having defined the sets above, we can define three types of semantic quality, namely of descriptive and prescriptive models. First, for descriptive models, it will have the same meaning as 'semantic quality' in the previous framework (validity: $M/D=\emptyset$, completeness: $D/M=\emptyset$). Second, prescriptive models will be compared to D^O instead of *D*, that is, validity: $M/D^O=\emptyset$, completeness: $D^O/M=\emptyset$. Third, there is the relationship between *K* (the current knowledge) and *M* (the model). In the previous framework, this was called 'perceived semantic quality' since it was the only kind of semantic quality that was possible for the participants to evaluate (since nobody can have complete and accurate knowledge of the domain *D* for anything but very limited toy problems). However, it can be argued that a quality that cannot be evaluated is not very interesting – as argued in Eliason & Ågerfalk (2003), people can neither be satisfied nor dissatisfied with what they cannot perceive. Hence, we change the terminology for the revised framework: what was previously called 'perceived semantic quality' is now called just *semantic quality*, which fits better with the view that reality is socially constructed (Dahlbom, 1991). The relationships of *M* with *D* and D^O are instead called *ideal semantic quality* (descriptive and prescriptive), indicating that these qualities are beyond evaluation and human achievement. One might of course discuss whether it is of interest to define such concepts at all. But the alternative would be to remove the domain (*D* and D^O from the framework altogether, and then the framework would not illustrate equally well that the participants' knowledge of the domain might be incorrect). As it stands now, the framework is open for the possibility that a model might have a high semantic quality (i.e., everyone thinks it is a good model), yet a low ideal semantic quality (i.e., everyone has misunderstood something).

The most important change is still that pragmatic quality is made more ambitious than mere understanding, it can now be decomposed into several different concerns:

- *Overall learning*: Let ΔK^M be the increase of the set *K* (i.e., the current knowledge) facilitated by the model *M*. Then, the overall learning gain of the model is $\Delta K^M \cap K^N$, that is, the new knowledge acquired by the organisation, which is also within the knowledge need. Similarly, let ΔK^m be the increase of the set *K* caused by the modelling *activity*, a similar knowledge gain $\Delta K^{m} \cap K^{N}$ may be associated with this.

- *Local learning*: Often the goal of a modelling activity might be knowledge transfer, that is, there is one, or are few persons, in the organisation who know something, but there is a wish to make this knowledge available to more people: $((K_i/K_j) \cap K_j^{N})$, that is, there is knowledge held by a person or group in the organisation, but not by person/group *j*, although it is within the knowledge need of *j*. Similar to above, the improvement will be $\Delta K_j^{M} \cap K_j^{N}$, the added knowledge of *j*, which was within *j*'s knowledge need.
- *Overall domain improvement*: Let ΔD^{M} be the change of the domain *D* facilitated by the model, and ΔD^{m} similarly by the modelling activity. The improvement resulting from the model and modelling activity together will then be $(\Delta D^{M} \cup \Delta D^{m}) \cap D^{O}$, that is, the alterations to the domain, which were in line with the optimal domain. Local domain improvement (i.e., for a particular person or group, but not necessarily for the entire organisation) could be defined in the same manner as local learning quality.

For an active model, a change to this model might cause a direct change of the domain, hence it might be reasonable to call this a *model-act* (analogous to speech-act); this is the relationship between modelling activity and domain change. In other cases, the modelling activity as such might not bring about dramatic changes to the domain, but later efforts based on the model may do so – this is more the traditional situation where the model is the basis for the development of a new information systems. Thus, the increased focus on the various activities (ovals) that were not present in the original SEQUAL diagrams has made it easier to see more possible benefits of modelling and classify these more precisely, as well as emphasising modelling as action in a way that was not so clear in the original framework.

Applicability of the new framework: maritime industry case

As a preliminary investigation into how the extensions of the framework is useful as for assessing and understanding the use of models, we have applied this relative to a concrete industrial project situation. We have earlier used the original framework in the assessment of this case, as reported in Krogstie *et al*. (2004).

The case organisation has offices spread around the world. The business rules and work procedures for the core business were established centrally. Around 20 years ago, small independent initiatives on providing certification services started up in some of these offices. The certification activity within the different units soon increased. In contrast to the traditional core business within the organisation, certification was not initiated from the main office, but each certification unit was developed based on local needs and procedures. Each certification unit developed their own systems and work procedures for back-office and support activities, such as marketing, sales, planning, issuing certificates, etc. Some units also developed their own software applications. From the mid-1990s, a common software application was locally implemented in some units. Over the years, each unit personalised this tool to their local needs. In 2000, certification centrally decided to focus on improved efficiency, and chose to start the work on harmonising the work processes. The harmonised work processes should be standardised processes, with the possibilities of local adaptation. This way, certification is now implementing centrally based processes, in a business that is used to work according to local procedures. A new software application to support the harmonised work processes is currently being implemented throughout the organisation.

The project has actively used modelling, both in developing the harmonised processes and in communicating them to different types of audience. The project has reused knowledge as well as models from earlier process model activities within the business area, and the harmonised work process models are also being further used in other, later initiated projects within the organisation.

The work processes within the business area were first modelled at a high level in the mid-1990s as a part of an official document describing the business. These models were created in a spreadsheet (Excel), but have later been implemented in the modelling tool Visio using IDEF0 process modelling.

In 2000, it was decided to look into possible changes in the work processes within the business area, how these were developed and adapted locally, and what could be improved. A pre-project went through several workshops involving partners from different offices within the network, where domain experts from the business area worldwide defined a list of areas to improve in the current work processes. The pre-project concluded on starting an efficient and harmonising project, in order to first develop an ideal process, and then a harmonised process, and also looked into the possibility of information system support.

The 'ideal' work processes was described. When defining the harmonised process, technical and economical constraints have influenced the models. This modelling was performed by the process developer, and the target audience of the models were domain experts. The results of this process were textual descriptions and high-level models of the harmonised processes.

An information systems development project was established in 2001. The process models were initially used by the process developer to communicate with the software engineers. In order for the harmonised processes to be implemented worldwide, key personnel from the business area management went to the different local units, analysing their local work processes in relation to the harmonised processes (gap analysis). The IDEF0 models of the harmonised processes were actively used during the analysis process.

The developed system was structured according to the harmonised process. Local implementations could adapt these to a certain degree to fulfil local needs. The process models were also used to teach people about both the processes and tool to support the processes. In addition, they were to be used as a basis for process improvement. Further details on the case study and how this was conducted can be found in Krogstie *et al*. (2004).

When analysing this in Krogstie *et al*. (2004), we used the framework of Sedera *et al*. (2003) as an overall structure. They have presented a process modelling success model in which the identified success measures are as follows:

1. *Model use*: How extensively the models are applied and utilised.
2. *User satisfaction*: The extent to which users believe process modelling meets the fulfilment of the objectives that underlay the modelling project.
3. *Process impact*: This measures the effects of modelling on the process' performance.
4. *Process model quality*: The extent to which all desirable properties of a model are fulfilled to satisfy the needs of the model users.

Trying to use this framework, we quickly discovered that their discussion is to a large extent based on modelling in connection to the development of ERP systems in a traditional organisation, and that in our case, a more detailed model was needed to explain all the relevant aspects for a networked organisation. When detailing this, we were able to apply the previous quality framework to look more closely at process model quality. On the other areas, we did not get support from the original framework. On the other hand, looking at the new framework, we find that it helps understanding areas of process impact and model use, by exactly looking at the extended understanding of pragmatic quality relative to how the model impacts the other sets of the framework. The area of user satisfaction is found to be a mean for actually getting model use and process impact, and we restructure our learning as reported in the previous paper, especially according to change of domain and change of knowledge below.

Change in domain *D*: The IDEF0 models in our case study have been used for communication for different purposes within the business area. Examples are the use of the models as a communication artefact between domain experts in order to improve the work processes. The consequences of implementing the harmonised processes are that all units have to change their work processes somewhat, supported by a new information system. One informant claimed that it was not the models that changed the work processes in the unit, but the process of performing the modelling. The models acted as a very important artefact to focus the discussions around. The representative of the IT department claimed that the process models had created 'a fantastic foundation for the participants in the project', and that they were important to be able to change the work processes in the business area by establishing a common understanding across the different units.

Change in knowledge *K*: The swimlane models were used for the software developers to understand the domain. The IDEF0 models were also used as a tool to perform the initial gap analysis in the units. For rollout purposes, the modified swimlanes were used in order to both reveal the gaps, and train the users. The IDEF0 models were also used as a basis for developing the user interfaces of the information system. The work process models were reflected through the menu structure in the information system, and by articulating the models in this way.

We do not reiterate our analysis of the model quality (Krogstie *et al*., 2004).

Conclusions and future work

In this paper, we have discussed the shortcomings of the semiotic framework for model quality (SEQUAL), especially in the context of process models and active models. On the basis of this, two main new contributions have been made:

- a discussion on the dynamic interplay between the various sets of SEQUAL in the light of interactive processes (Figure 2);
- AN outline of a revised framework (Figure 3) where the notion of quality for a model is extended by looking at its ability to facilitate learning and action, more than just being a representation of a domain.

In particular, the revisions have solved the following problem:

- Distinguishing between the actual domain *D* and the optimal domain D^O has made it easier to discuss clearly the quality of requirements generally (i.e., a prescriptive model of a future 'to-be' situation) as well as of descriptive models of the current 'as-is' situation, and of process improvement specifically. Also, a change of terminology has moved to the front the kind of semantic quality that can possibly be evaluated, while the relations that are beyond human evaluation are clearly signalled to be so.
- Distinguishing between the current knowledge *K* and the knowledge need K^N has similarly made it easier to arrive at a fruitful definition of knowledge quality, which can now be considered relative to this need, rather than relative to the domain as such.
- Pragmatic quality has been redefined to achieve increased importance. Rather than dealing with the participants' mere understanding of the model, it now addresses the model's ability to facilitate learning (i.e., increasing *K*) and action (i.e., changing *D*).
- In addition to considering such pragmatic quality just for the model, it is also considered for the *modelling activity*. This makes it possible to assess fairly modelling

efforts, which end with little formal documentation, or where the developed models are simply thrown away, but where the modelling effort still contributes significantly to increase the participant's knowledge and thereby facilitate action.

What the revision does not fix is the framework's disability to facilitate precise, quantitative evaluations of models – it is still focussed on qualitative evaluations and on the understanding of quality in modelling more than being an instrument for scoring of models. But in this respect, the revised framework is simply in line with the intention behind the original framework.

For further work, it will of course be interesting to test out the revised framework in additional case studies. And even if the framework is qualitative rather than quantitative, it might still be interesting to pursue investigations on extending it with metrics for the various quality types and to embed these metrics in tool support such as, for example, the combination of METIS/Workware developed in EXTERNAL (Krogstie & Jørgensen, 2004).

About the authors

John Krogstie holds a Ph.D. (1995) and an M.Sc. (1991) in Information Systems, both degrees from the Norwegian University of Science and Technology (NTNU). He is a Professor in Information Systems at IDI, NTNU, Trondheim, Norway. He is also a senior advisor at SINTEF. He was employed as a manager in Accenture, 1991–2000. John Krogstie is the Norwegian Representative for IFIP TC8 and Vice-Chair of IFIP WG 8.1 on Information Systems Design and Evaluation, where he is the initiator and leader of the special interest group on Mobile Information Systems. He has published around 75 refereed papers in journals, books and archival proceedings since 1991.

Guttorm Sindre holds a Ph.D. in Computer Science ('databehandling'), Norwegian Institute of Technology (NTH), University of Trondheim, September 1990 and a *Master of Engineering* ('*sivilingeniør*') in Computer Science ('databehandling'), Norwegian Institute of Technology (NTH), University of Trondheim, February 1988. From 1992 to 1995 he was an associate professor in Software Engineering, and from 1999–2003 an associate professor in Information Systems at IDI, NTNU. In 2003 he was promoted as professor. He has guided several Ph.D. students. He has written 10 papers in journals and books and more than 50 papers in international conferences and workshops since 1987. For the study year 2002/03, he was on a sabbatical visit to the University of Auckland, New Zealand. He is currently manager of the five MNOK research project WISEMOD (2004–2007), funded by the Norwegian Research Council over the IKT-2010 programme, and is also involved in the EU IST 6FP NoE project INTEROP.

Håvard D. Jørgensen holds a Ph.D. (2004) and an M.Sc. (1993) in Information Systems, both degrees from the Norwegian University of Science and Technology (NTNU). He is currently a senior researcher at Troux Technologies. He has earlier been working as a research scientist at SINTEF ICT, which is part of Scandinavia's largest research institute. After some years in industry, he joined SINTEF in 1996. His main areas of interest are flexible workflow systems, groupware and knowledge management. He has published around 25 refereed papers in journals, books and archival proceedings.

References

ÅGERFALK PJ (2003) Information systems actability: understanding information technology as a tool for business action and communication. PhD Thesis, Department of Computer and Information Science, University of Linköping, Sweden.

ÅGERFALK PJ and ERIKSSON O (2003) Usability in social action: reinterpreting effectiveness, efficiency, and satisfaction. In *Proceedings of European Conference on Information Systems (ECIS'03)* 19–21 June, Naples, Italy (CIBORRA C, MERCURIO R, DE MARCO M, MARTINEZ M and CAVIGNANI A, Eds), AIS: Atlanta, USA.

ARGYRIS C and SCHÖN D (1973) *Organisational Learning: A Theory of Action Perspective*. Addison Wesley: Reading, MA.

BOLAND RJ and TENKASI RV (1995) Perspective making and perspective taking in communities of knowing. *Organization Science* **6(4)**, 350–372.

BØVING KB and PETERSEN LH (2002) Design for dummies: understanding design work in virtual workspaces. In *Proceedings of PDC2002*, 23–25 June, Malmö, Sweden (BINDER T, GREGORY J and WAGNER I, Eds), CPSR: Palo Alto, CA, USA.

BRAF E (2004) Knowledge demanded for action: studies on knowledge mediation in organizations. PhD Thesis, Department of Computer and Information Science, Linköping University, Sweden (Linköping Studies in Information Science, Dissertation no. 10).

BRASETHVIK T and GULLA JA (1999) Semantically accessing documents using conceptual model descriptions. *Proceedings of the First International Workshop on Conceptual Modeling and the WWW*, Lecture Notes in Computer Science, Vol. 1727, Paris, France, pp 321–333, Springer Verlag, Berlin.

CARLSEN S, KROGSTIE J, SØLVBERG A and LINDLAND OI (1997) Evaluating flexible workflow systems. In *Proceedings of Hawaii International Conference on System Sciences (HICSS-30)*, Maui, HI (NUNAMAKER J and SPRAGUE R, Eds), IEEE CS Press: Los Alamitos, CA.

CARLSEN S and GJERSVIK R (1997) Organizational metaphors as lenses for analyzing workflow technology. Paper presented at the *GROUP'97 Conference*, Phoenix, AZ, 16–19 November.

COOK SDN and BROWN JS (1999) Bridging epistemologies: the generic dance between knowledge and knowing. *Organization Science* **10(4)**, 381–400.

DAHLBOM B (1991) The idea that reality is socially constructed. In *Software Development and Reality Construction* (FLOYD C, ZULLIGHOVEN

H, Budde R and Keil-slawik R, Eds), pp 101–126, Springer Verlag: Berlin.

Eliason E and Ågerfalk PJ (2003) The D.EU.PS Model: a tool for studying effects of information systems design. In *Action in Language, Organisations, and Information Systems (Proceedings of ALOIS 2003)* (Goldkuhl G *et al*., Eds), pp 191–205, Linköping: Sweden.

Farshchian BA (2000) Gossip: an awareness engine for increasing product awareness in distributed development projects. In *Advanced Information Systems Engineering – Proceedings of CAiSE 2000*, Lecture Notes in Computer Science, Vol. 1789 (Wangler B and Bergman L, Eds), Springer Verlag: Berlin.

Goldkuhl G (2002) Anchoring scientific abstractions: ontological and linguistic determination following socio-instrumental pragmatism. *European Conference on Research Methods in Business and Management*, Reading, UK, 29–30 April.

Goldkuhl G and Ågerfalk PJ (2002) Actability: a way to understand information system pragmatics. In *Coordination and Communication Using Signs* (Liu K *et al*., Eds), pp 85–114, Kluwer Academic Publishers: Dordrecht, The Netherlands.

Greenwood RM, Robertson I, Snowdon RA and Warboys BC (1995) Active models in business. *Fifth Conference on Business Information Technology, CBIT'95*. Department of Business Information Technology, Manchester Metropolitan University: UK.

Hjalmarsson A and Lind M (2004) Managing the dynamic agenda in process modelling seminars: enhancing communication quality in process modelling. In *Proceedings ALOIS*2004*, 17–18 March (Goldkuhl G, Lind M and Cronholm S, Eds), Linköping University: Linköping, Sweden.

Jørgensen HD (2004) Interactive process models. PhD Thesis, Department of Computer and Information Science, Norwegian University of Science and Technology, Trondheim, Norway (Antoni O, Masatoshi Y, Eric Y, Eds), Springer Verlag: Berlin, Germany.

Krogstie J (1995) Conceptual modelling for computerized information systems support in organisations. PhD Thesis, University of Trondheim, Norway.

Krogstie J and Jørgensen HD (2002) Quality of interactive models. In *First International Workshop on Conceptual Modelling Quality (IWCMQ'02)*, 11 October 2002. (Olive A, Yoshikawa M and Yu E, Eds), Springer Verlag: Berlin, Germany.

Krogstie J, Lindland OI and Sindre G (1995) Defining quality aspects for conceptual models. In *Proceedings of the IFIP8.1 Working Conference on Information Systems Concepts (ISCO3): Towards a Consolidation of Views*, 28–30 March, Marburg, Germany (Falkenberg E, Hess W and Olive E, Eds), Chapman & Hall: London, UK.

Krogstie J and Jørgensen HD (2004) Interactive models for supporting networked organizations. In *Proceedings of CAiSE'2004*, June 9–11, (Persson A and Stirna J, Eds), Springer Verlag: Berlin, Germany.

Krogstie J, Dalberg V and Jensen SM (2004) *Harmonising Business Processes of Collaborative Networked Organization Using Process Modelling in PROVE'04*, Toulouse, France.

Lindland OI, Sindre G and Sølvberg A (1994) Understanding quality in conceptual modeling. *IEEE Software* **11(2)**, 42–49.

Moody DL, Sindre G, Brasethvik T and Sølvberg A (2002a) Evaluating the quality of process models: empirical analysis of a quality framework. In *Proceedings of the 21st International Conference on Conceptual Modeling* (ER'2002), 7–11 October, Tampere, Finland (Spaccapietra S, March ST and Kambayashi Y, Eds), Springer Verlag: Berlin, Germany.

Moody DL, Sindre G, Brasethvik T and Sølvberg A (2002b) Evaluating the quality of information models: empirical testing of a conceptual model quality framework. In *Proceedings of the 25th International Conference in Software Engineering (ICSE'03)*, Portland, OR, USA, 3–10 May 2003 (Kazman R, Bass L and Bosch J, Eds), ACM Press: New York, USA.

Nonaka I (1991) The knowledge-creating company. *Harvard Business Review* **69**, 96–103.

Polanyi M (1966) *The Tacit Dimension,*. Routledge & Kegan Paul: London.

Sedera W, Rosemann M and Doebli G (2003) A process modelling success model: insights from a case study. *11th European Conference on Information Systems*, Naples, Italy.

Stamper R (1996) Signs, norms, and information systems. In *Signs at Work* (Holmqvist B *et al*., Eds), pp 349–397, Walter de Gruyter: Berlin, Germany.

Voss A, Procter R and Williams R (2000) Innovation in use: interleaving day-to-day operation and systems development. In *Proceedings of the CPSR/IFIP WG 9.1 Participatory Design Conference*, November 28–December 1, New York, NY, USA (Cherkasky T, Grecnbaum J, Mambrey P and Pors JK, Eds), CPSR: Palo Alto, CA, USA.

Walsham G (2004) Knowledge management systems: action and representation. In *Proceedings ALOIS*2004*, 17–18 March (Goldkuhl G, Lind M and Cronholm S, Eds), Linköping University: Linköping, Sweden.

Walsham G (2005) Knowledge management systems: representation and communication in context. *Systems, Signs and Actions* **1(1)**, 6–18.

Wegner P (1997) Why interaction is more powerful than algorithms. *Communications of the ACM* **40(5)**, 80–91.

Wegner P and Goldin D (1999) Interaction as a Framework for Modeling. In *Conceptual Modeling. Current Issues and Future Directions* Lecture Notes in Computer Science, Vol. 1565 (Chen PP, Akoka J, Kangassalo H and Thalheim B, Eds), Springer Verlag: Berlin, Germany.

Zuboff S (1988) *In the Age of the Smart Machine*. Basic Books Inc.: USA.