

INCREASING THE LEVEL OF AUTOMATION IN ORGANISATIONS: SOME REMARKS ON FORMALISATION, CONTINGENCY AND THE SOCIAL CONSTRUCTION OF REALITY

"We want to establish an order in our knowledge of the use of language: an order with a particular end in view; one out of many possible orders; not the order."

Ludwig Wittgenstein

"Language is a social art."

Willard van Oman Quine

Ulrich Frank
Institut für Wirtschaftsinformatik
Universität Koblenz-Landau
Rheinau 1
56075 Koblenz
Germany

Key Words: Language, model, research goal, research strategy, evaluation of artefacts.

Abstract

Concepts and theories to foster automation in organisations can be regarded as an essential research goal of the information systems discipline. Based on a few assumptions about essential inhibitors of successful automation, this paper describes and analyzes three prototypical approaches to accomplish a higher level of automation in organisations: a reductionist, a inductivist, and a constructivist approach. While there is evidence that the constructivist approach is most effective, it will be shown that it implies severe epistemological problems. It will be argued that there is hardly any straightforward solution to these problems. However, for the information systems discipline to establish a profile of its own, it has to accept them as a pivotal challenge.

1 Introduction

The information systems discipline is still characterised by a remarkable diversity. Among other things, it includes research topics such as the economic evaluation of information systems (for instance: Ragowsky et al. 1996, Saarinen 1996), the prerequisites of creativity in information systems organisations (Couger 1994), the development of information models and even the investigation of sexual harassment via e-mail (Sipior & Ward 1997). This diversity is accompanied by a number of different research methods. While this "so-called free-for-all situation" (Banville & Landry 1992, p. 87) has caused various authors to emphasise the need for a coherent profile of the discipline (Banville & Landry 1992, Hirschheim 1992), there is one implicit topic that is hardly questioned as an essential orientation of information systems research: the development of frameworks or theories that are suited to improve efficiency by increasing the level of automation in organisations. In this paper we will analyse three prototypical approaches to achieve this goal: a reductionist, an inductivist, and a constructivist research strategy. The results of this investigation suggest that a constructivist approach - which is quite common for the actual evolution of corporate information systems - is most promising. However, at the same time it is accompanied by severe epistemological challenges for the information systems discipline. We will argue that these problems can hardly be solved in a satisfactory way. Facing them, however, recommends to reflect and eventually change the way information systems research is organised.

2 The Reductionist Approach: Formalizing Individual Knowledge

Lifting the level of automation in organisations suggests a concentration of effort on tasks which cannot so far be performed by machines. Since those tasks are performed - more or less successfully - by human beings, the automation of individual intellectual capabilities would allow to accomplish a higher level of automation in organisations, too. In the context of this paper, we call a research strategy reductionist if it focuses on the automation of individual competence. The classical example of a reductionist approach is the strategy that is inherent in Artificial Intelligence. It aims at reconstructing human intelligence or relevant parts of it with machines. If it is successful, such a strategy would allow for "... substituting machines for any and all human functions in organizations." (Simon 1964, p. 594). We cannot decide whether or not it is possible to achieve this ambitious goal. Therefore, it is not useful to discuss this possibility. However, the lack of success in the respective research has dampened the early enthusiasm. As a consequence, a new, less ambitious approach emerged in the eighties: Instead of reconstructing human intelligence, the "expert system" approach focused on the identification and formalisation of *knowledge* which can be interpreted by a machine in order to automate intellectual tasks like diagnosis, planning and decision making in general. Although expert systems failed to fulfil the daring promises made in the eighties, formalisation of knowledge is still a pivotal strategy in fostering automation and supporting efficient decision making. Within the information systems domain, *managerial* decision making is of special importance. In the following section, we will analyse possibilities for and limits of automating managerial decision making through the formalisation of managers' knowledge.

2.1 Formalizing Managerial Decision Making: The Pivotal Challenge

The variety of tasks that are performed by managers is immense. Some of those tasks have been automated already, many others seem not to allow for automation. For the purpose of our investigation, it would be sufficient if there was an essential, common ingredient of managerial competence that, at the same time, resisted to being formalised. Usually, a manager has to act in an environment that is characterised by imponderables, ambiguity and risk. Therefore, it seems reasonable to identify the ingredient we are looking for as the ability to cope with *contingency*. The term contingency denotes aspects of reality - like events or processes - which we cannot be certain about: We do not know for sure when they will occur or what outcome they will produce. Therefore, acting in a contingent environment implies to select from options we cannot take for granted: They are possible, but not necessary, or - as Luhmann (1984, p. 152) puts it: The notion of contingency denotes something given (experience, expectations, thoughts) with respect to its possible "Anderssein" (being different).

If dealing with contingency is a core task of managers, the question remains what it takes to act successfully in a contingent environment. Instead of analysing this question directly, we assume that the basic assumption of the reductionist approach is appropriate: Professional decision making under contingency is usually based on knowledge which can be formalised. Additionally, we simplify our investigation by avoiding the ontological question about the nature of human knowledge. Instead, we focus on formal representations of "knowledge" only. This is feasible because we are primarily interested in the question whether these representations are sufficient for successful automation.

2.2 Remarks on "Vague" Knowledge

The representation of knowledge for the purpose of automation requires a formal language. Different from a natural language, a formal language is a system of symbols that is generated by a calculus. It is based on a well defined alphabet of characters and a clear syntax. The semantics of propositions expressed in a formal language is provided by an unambiguous interpretation.¹ In traditional logic, an interpretation defines whether a proposition is true or false. For this reason, it seems that formal languages are not appropriate to describe knowledge about contingent domains: It is an essential feature of those domains that we are not certain about all their relevant aspects. At first sight, a logic that allows to express uncertainty - like probabilistic logic (Nilsson 1986) or Fuzzy logic (Zadeh 1975) seems to be well suited to represent so called vague knowledge about contingent domains. However, there are severe objections against the use of such representations for the automation of intellectual tasks. The degree of certainty (or membership) will usually be assigned by human judgement. Contingency means that we do not know exactly how reliable our assumptions are. Assigning a number nevertheless implies to deal with a paradox in the end: certainty about uncertainty. Moreover, such an approach suffers from the fact that most people are not qualified to deal with probabilities. A plethora of research on judgement under uncertainty has revealed numerous sources of bias (for instance: Kahneman et al. 1982, Hogart 1980): "... people systematically violate the principles of rational decision making when judging probabilities ..." (Slovic et al. 1977, p. 169).

2.3 Challenges and Limits of Automation via Knowledge Representation

Although the problems to be expected from representing vague knowledge dampen the hope for knowledge based systems that allow to automate decision making under contingency, they are not sufficient to compromise the basic assumption of the reductionist approach: The human ability to deal with contingency is based on knowledge which, in principle, can be formalised. A well known objection against this assumption claims that there is knowledge that we simply cannot express - not to speak of formalise it. Wittgenstein uses a metaphor to illustrate this argument: We know how high the Mont Blanc is - and we are able to describe this knowledge in a comprehensive way. This is, however, different with our knowledge about the sound of a clarinet (Wittgenstein 1971, §63). While this argument - "knowing but not being able to tell" - certainly denotes a severe challenge, it does not exclude the possibility of retrieving knowledge that seems to be tacit.

Judgement in the face of contingency in social systems requires to take into account a plethora of information. While some of this information may have a clear meaning (like revenues, profits etc.), other information requires additional interpretation (like information about the performance of employees or the trustworthiness of customers). Often, it will be necessary to actually *perceive* aspects of reality in order to judge them properly. Interface devices that are able to monitor reality - like cameras or microphones - are certainly not sufficient since they do not detect any semantics. Even software that allows for the analysis of natural language input would not be of much help, since it completely neglects the intention of the speaker and the background of his statements. However, in order to make appropriate decisions in a contingent environment, it is essential to judge a speaker's confidence in his statements and whether or not he may be lying.

These thoughts have a severe impact on the basic assumption of the reductionist approach

1. Notice that there may be more than one interpretations. An interpretation is sometimes called a model.

since they suggest that knowledge alone is not sufficient for coping with reality. Moreover, there is evidence that knowledge and perception cannot be separated: Knowledge is a prerequisite for selective perception, perception in turn is necessary not only to gain but also to apply knowledge. Simon (1983, p. 28) speaks of a "feedback tie", according to Weick (1980, p. 159) "Action, perception, and sense-making exist in a circular, tightly coupled relationship ...".

Today, we know too little about the physiological and intellectual capabilities that constitute perception to allow for their automation. Nevertheless, similar to the basic assumption underlying early research in Artificial Intelligence, we cannot exclude the possibility. There is, however, another aspect of managerial decision making that denotes a definite limit of the reductionist approach. There is evidence that managers sometimes make decisions without substantial knowledge. If there is no knowledge, there is not chance to use it as a basis for automation. While this is not surprising, it is important to note that humans can nevertheless be successful in coping with contingency. Since managerial competence is "hard to judge" (Perrow 1986, p. 11), selecting managers is based on "beliefs" (Pfeffer 1978, p. 24). Therefore, their judgements - even if they are not based on substantial knowledge - may be commonly accepted as an appropriate orientation. Hence, managers can be successful by constructing "'objective' features of their surroundings" (Weick 1980, p. 164). Self-fulfilling prophecies are an obvious example. In other cases - shared values, language games - the influence is more subtle. There is no doubt that this symbolic function of decision making under contingency does not allow to be substituted by machines: It is essentially related to social interaction.

3 The Inductivist Approach: Searching for Generic Patterns of Information Use

The limits encountered for the reductionist approach do not compromise it in general: There is certainly still individual knowledge that can be formalised. Even if it does not allow to completely automate decision making, it may help with increasing the level of automation in organisations by *supporting* human decision making. However, we should not expect a substantial shift in the level of automation if we focus on the reductionist approach only. While the reductionist approach is focusing on the automation of individual skills, the following approach is less ambitious. It tries to discover common patterns of existing information use. In order to stress this aspect, we call it the inductivist approach although it is not necessarily based on induction as a research method. Note that this is a prototypical strategy that one can hardly find in a pure sense.

3.1 Prospects

Beside the difficulties to automate intellectual capabilities to cope with contingency, one major inhibitor of automation in organisations is the lack of common concepts to represent and process information. There is an immense diversity in information concepts that are used within software and that are relevant for information interchange. Almost any software used in organisations is based on its own concepts of entities like "customer", "address", "invoice", "order", etc. Different from the reductionist approach, formalising those concepts is not a substantial problem. Nevertheless, the variance of existing concepts hinders automation. On the one hand, it imposes a problem for information exchange (hence: for the *integration* of information systems). If the receiver of a message uses a concept that includes more information than the concept used by the sender, a transformation will either be accompanied by the lack of information or by the risk that is related to the reconstruction of missing information. Even in the case that there are no semantic differences, an automatic transformation may fail because it is too expensive - in the case of n participants with individual concepts, there is need to

implement $n(n - 1)$ transformations. On the other hand, the lack of common concepts hinders the *reusability* of software. With increasing levels of software reuse, the cost per copy is decreasing. This would in turn open up new areas of economic automation.

The idea of an inductivist approach is based on the assumption that there are common patterns of information use. Discovering them would allow to establish reference concepts which would decrease transaction costs and foster software reuse.

3.2 Feasibility and Pitfalls

The variance of requirements and of actual information use indicates that it would be a remarkable challenge to find concepts that are suitable for all organisations. This is especially the case for concepts that incorporate a high level of semantics (information content): The higher the information content of a concept the less the probability that it represents a particular facet of reality in an adequate way. In principle, however, it would be possible to find generic concepts, since - at a point in time - it is a finite number of cases that has to be taken into account. In order to cover a high amount of variance, those concepts would have to be very complex and therefore hard to implement and use. Without any doubt, identifying suitable generic concepts would cause tremendous costs. Independent from the challenges an inductive strategy would have to face, there are reasons why its results - if it would ever be successful - were of doubtful use. Actual information use is an expression of the ways tasks and processes are organised. Usually, (business) processes as well as the documents that they use have not been designed - not to speak of optimized - with the capabilities of modern information technology in mind. In order to fully exploit the potential of this technology, it is more promising to rethink traditional ways of performing processes and using information, instead of simply reproducing them. Schank (1985, pp. 23) uses an analogy to illustrate this thought:

"The first users of cars and computers had to struggle to make these completely new machines operate within the limits of the systems that were designed for an earlier world. ... The computer industry's perspective suffers from the same lack of creativity and long-term vision. Computers are severely limited by the world views and ideas that have preceded them."

Considering both, the tremendous effort that it would cause and the doubtful outcome, a purely inductivist - we could also say: a descriptive - approach to increase the level of automation in organisation is hardly an attractive option.

4 The Constructivist Approach: Preparing Reality for the Information Age

There is not doubt that suitable common concepts - we could also speak of a *common language* - would help to increase the level of automation in organisations. If we want to avoid the pitfalls imposed by replicating existing information use, we have to design new, artificial concepts that take into account the special advantages and restrictions of information technology. We call this approach constructivist to express that it aims at creating new shapes of reality. This is the case for two intertwined reasons. First and more obvious, the approach includes the analysis and possible redesign of processes performed in organisations. Second, the introduction of new concepts to describe reality will influence the way we perceive and speak about the affected reality.

The brief history of information technology offers numerous examples of constructivist approaches: The definition of character codes as well as many standards that define the representation and semantics of digitised information. However, so far those efforts have been mainly concentrated on concepts that incorporate little semantics. Only recently,

there are attempts to standardise high level concepts as well - like so called business objects (Object Management Group 1996). Standardisation processes are dominated by standardisation organisations and powerful stakeholders like governments and large corporations. In the following section, we want to look at the constructivist approach as a *research* strategy which has a number of epistemological implications that are less relevant for pragmatic approaches.

4.1 Reducing Contingency

Similar to the inductive approach, the constructivist approach aims at fostering integration and reusability of information systems. Different from the inductive approach, actual ways of using and producing information are not taken for granted. Instead, a constructivist strategy is based on the assumption that the effective exploitation of the potential provided by modern information technology recommends to redesign traditional means of communication and cooperation. In other words: It recommends to construct reality by introducing new ways of coordinating cooperative work. Where the inductivist approach assumes that the variance in using information and expressing it through languages is a necessary reflection of the variety of tasks to be taken into account, the constructivist approach relies on the presumption that variance in actual information use and related coordination mechanisms is the result of an - at least partially - arbitrary process. For this reason, reducing variance by introducing new common concepts to handle information would not necessarily cause dysfunctional effects. Moreover, if the processes they are to be used in were thoroughly designed, they would contribute to more efficiency. In any case, those common artefacts would allow for a high level of integration and reusability. There are numerous examples where concepts/artefacts were constructed to exploit the potential of computerised information systems rather than merely mapping existing concepts. For instance, languages and architectures to describe electronic documents, such as ODA, SGML, HTML (Appelt 1991, Goldfarb & Rubinsky 1990), and especially OpenDoc (The Opendoc Design Team 1994) are based on a notion of document that is significantly different from the traditional notion.

Where the reductionist approach aims at reproducing human capabilities to cope with contingency, the constructivist approach would reduce contingency - in other words: variance, ambiguity, and risk by introducing common infrastructures for organisations. This is very similar to the traditional notion of organising which is characterised by Weick (1980, p. 3) "as a consensually validated grammar for reducing equivocality ...". In order to foster automation, the focus of the constructivist approach is on creating *formal* concepts to handle information. Concepts required to exchange and manipulate information would be provided on an appropriate level of semantics. Since it can hardly be expected that there is usually only one general concept that fits all organisations, a certain variety would have to be provided. In the end, a common formal business language would allow for the automation of processes that cannot be automated in an economic way today.

4.2 Feasibility

The constructivist approach requires to specify information concepts on a high level of semantics (a common concept of an integer may be necessary but is certainly not sufficient). In order to define the meaning of a concept used within an application domain, we need an appropriate representation of this domain: Who is using a concept for what purpose? What is the relevant context (associated concepts, related processes etc.)?

The formal or at least: semi-formal specification of these concepts could be accomplished in any of many (semi-) formal languages. The representations should be intuitive for everybody who participates in the organisation of a domain. At the same time, it should provide an appropriate input for developing software. *Conceptual modelling* is regarded as the best option to combine both goals. In recent years, a number of authors (Katz 1990, Sowa & Zachman 1992, Petrie 1993) have suggested enriched modelling frameworks - often named "enterprise modelling" (a term which, however, is not used in a unique way). Such approaches usually suggest a number of views on the enterprise and intend to capture the relationships between these views. The development of enterprise models seems to be an appropriate strategy to stress the constructivist approach, since they combine the design of information systems and the related tasks of analysing and (re-) designing a firm's organisation and strategy.

Among other things, a method for enterprise modelling consists of a set of modelling languages that are used to model different views of an organisation - for instance: a static information model (like an object model), information exchange models (like message flow diagrams), models of business processes, etc. Multi Perspective Enterprise Modelling (MEMO), for example, offers a number of graphical modelling languages which are specified and integrated through metamodels which in turn are defined by concepts of a common meta-metamodel (Frank 1997, Frank 1998). A method for enterprise modelling could be used to develop generic reference models - for instance of certain processes in banks or in insurance companies together with concepts of the information needed within these processes. Meta concepts like specialisation or aggregation could be used to create more special models.

While the vision of generic enterprise models may be appealing, it is certainly a complex endeavour that raises the question whether it is feasible at all. At first sight, the obvious variety suggests that generic reference models can hardly satisfy the peculiarities of many organisations. Also, the high amount of time that it takes to develop reference models may cause them to be outdated as soon as they could be established. Indeed, a few attempts in the past - mainly focusing on reference models for manufacturing enterprises (Savage 1990, p. 136, ESPRIT 1991) were not successful from a pragmatic point of view: None of them has ever been implemented in a substantial number of organisations. However, there is evidence that organisations are able - and willing - to adapt their processes and information concepts to reference models. Vendors of so called standard business software - like SAP - emphasize that their software is based on optimized reference processes. For this reason, they recommend that their customers should adapt to those reference processes in order to gain competitive advantage. While it is not clear whether customers follow this recommendation in order to improve their processes or simply to take advantage of affordable standard software, it is a matter of fact that more and more organisations are adapted to reference systems.

5 Epistemological Problems

While any of the prototypical approaches we have discussed so far can be identified - either isolated or more likely combined - in actual efforts to increase the level of automation in organisation, we are looking at them as research strategies of the information systems discipline. Beyond the subtle differences between theories of knowledge¹, we propose that the results (typically: hypotheses or theories) a research

1. The most important theories of knowledge in Germany are "Kritischer Rationalismus" (for instance: Popper 1935, Albert 1968), "Kritische Theorie" (for instance: Habermas 1981), and "Konstruktivismus" (Lorenzen & Schwemmer 1975) - the first sometimes also referred to as "neo-positivistic", the latter as

approach produces should satisfy a number of requirements. Among other things, they should provide *general statements* about classes of entities, not statements about single entities, they should provide *new knowledge*, and they should provide *comprehensible procedures* to check them against criteria commonly regarded as relevant. This implies the ability to compare competing research results in a way that allows to select the most appropriate option.

Different from traditional goals of (empirical) research, the results of research related to our three research strategies are - at least in part - not just statements that can be checked by confronting them with reality (no matter what theory of truth one prefers). Instead, we are dealing with artefacts - knowledge bases and inference engines, information models, process models and may be corresponding software. While we cannot simply apply the criteria listed above to artefacts, we still assume that the creation of artefacts can be a research goal. The notion of scientific research is based on the idea of progress – in terms of growing knowledge and improving technologies. Progress, however, implies the existence of criteria that allow to discriminate between competing options – be it explanations of reality or artefacts that are to fit actual or future real world domains. A research discipline that does not seriously care about such criteria risks to sacrifice its identity.

Beyond the specific difficulties we have encountered for the reductionist approach, it seems that it does not satisfy the request for new knowledge since it aims at formalising existing knowledge. However, in principle the reductionist approach could lead to new explicit statements about reality by identifying and describing knowledge that was only tacit before. In this case, the evaluation of a knowledge base could be very similar to the evaluation of hypotheses or theories in science: Every proposition of the knowledge base would have to be confronted with reality. Although we do not propagate a naive realism, such an evaluation should usually lead to acceptable results. This is different, if a knowledge base contains propositions which are assigned a probability - especially if those "probabilities" are in fact thought to express a certain level of confidence. As we have already argued above, there is no entirely satisfactory way to evaluate probabilities. Evaluating the outcome only - like the answers given by a system - is not satisfactory either. If an answer itself is assigned a probability, its evaluation faces the same problem as the evaluation of uncertain propositions in the knowledge base. While testing how well the answers/recommendations with the highest probability actually work seems to be a pragmatic approach, it is hardly acceptable from an epistemological point of view: A set of "appropriate" answers can still be based on inappropriate propositions/probabilities, leaving the risk of failure with any additional try.

The inductivist approach tries to discover general concepts of information and general patterns of information use. This is very similar to traditional empirical research. The evaluation of results produced by this approach seems to be straightforward: General statements (about concepts and processes) have to be checked against reality by asking whether they are suited to describe concepts and processes in actual organisations in an appropriate way. If we abstract from the problems of checking statements about reality against reality, there is still another problem to be faced with the inductivist approach: In the end, it is to provide (semi-) formal artefacts (like conceptual models, specifications or software). Although those artefacts are based on statements about reality, they require a specific evaluation. In order to design (software) artefacts, we need to build abstractions -

"hermeneutic". In my opinion, the main difference between these methodologies can be seen in the procedures suggested to justify (or falsify) theories.

or models - of the real world. Those models should provide both, an intuitive representation of the real world domain they are directed at, and an appropriate basis for the implementation of software. Evaluating a conceptual model against these requirements is a delicate task. A model can be regarded as intuitive, if it corresponds to individual patterns of perception and conceptualisation. Those patterns, however, are not only difficult to identify, they also vary from person to person as well as over time. Software engineering requirements are often not coherent as well. Therefore, designing conceptual models usually includes decisions which are in part arbitrary. In recent years there has been growing awareness of those problems. There are a few publications that suggest criteria/measures for evaluating the quality of conceptual models (Lindland et al. 1994, Krogstie et al. 1995) - none of them being convincing in the end.

Moreover, developing conceptual models of real world domains imposes the challenge to evaluate modelling *languages*, since a modelling language (its semantics, abstract syntax and graphical notation) has a pivotal impact on the quality of models. Although we are able to reflect upon language, for instance by distinguishing between object and meta level language, our ability to speak and understand a language is commonly regarded as a competence that we cannot entirely comprehend (Lorenz 1996, p. 49). Therefore any research that either aims at analysing a language and its use or at inventing new "language games" (i.e. artificial languages and actions built upon them), has to face a subtle challenge: Every researcher is trapped in a network of language, patterns of thought and action he cannot completely transcend - leading to a paradox that can hardly be resolved: Understanding a language is not possible without using a language. At the same time, any language we use for this purpose will bias our perception and judgement – or, as the early Wittgenstein put it: "The limit of my language means the limit of my world." (Wittgenstein 1981, §5.6).

Such considerations may seem to be of philosophical nature only. However, they characterise precisely one dilemma that evolves from the evaluation of modelling languages. Evaluation implies the knowledge of a modelling language. The more you know about the requirements of modelling, the more likely will you be able to understand existing concepts or to point at missing features. Modelling languages are usually designed by people who have gained an outstanding experience with the use of such languages. Usually they will design a language to fit their preferences. While both, the design and the evaluation of modelling languages require modelling experience, it is exactly this experience that will have a tremendous influence on the outcome of those efforts. Not only that this sort of bias hinders the invention of new modelling paradigms, at the same time – and this is an additional problem – we can assume that many people who are affected by modelling languages have perceptions and cognitive styles different from the language experts.

The constructivist approach inherits all the epistemological problems encountered for the inductivist approach. In addition to them, it causes another tremendous problem. Different from the inductivist approach, it does not aim at representing parts of the real world. Instead, it aims at designing new ways of performing collaborative tasks and creating artefacts that fit those future worlds. The pivotal goal of (re-) designing the world is the reduction of contingency. It is hard to tell whether or not this goal has been accomplished, since reducing a system's contingency is always accompanied by a temporary growth of complexity (Luhmann 1967, p. 109) which in turn may cause additional contingency. For instance: If sources of ambiguity are eliminated from a system, somebody who was used to this sort of ambiguity may find the new precise concepts even more contingent as long as he has not understood their meaning.

The evaluation of artefacts has to face two challenges. First, any construct has to be evaluated against the basic assumption that it is suited to foster automation without causing side effects that would overcompensate for the benefits of automation. While this is difficult enough, it is not sufficient: There should also be a way to compare alternative suggestions to (re-) construct the world. Second, there are (implicit) value judgements: Those constructs will affect the way we handle and perceive information, the organisation of our work - and the language we speak. While ignoring individual interests is hard to accept, identifying them in a reliable way (humans are able to learn) is hard to accomplish.

In this situation, one could leave the evaluation to evolution: Those alternatives are most suitable that survive/dominate in the end. Although this "best practice" approach to evaluation is rather common in the information systems domain, it is no satisfactory option. First, it does not allow for an ex ante evaluation which is desirable because the realisation of a particular design only for the purpose of testing it will usually be no option. Second, and more important, best practice means to sacrifice scientific standards for criteria which are common (and maybe appropriate) for markets. At the same time, emphasising the need for scientific evaluations of artefacts has to face objections, too. On the one hand, it can easily be mistaken for a positivistic attitude: prescribing those ignorant practitioners what language is best for them. On the other hand, considering the little influence the information systems discipline takes these days on the actual evolution of information systems, there seems to be hardly a chance to propagate artefacts produced by the discipline in practice.

Unfortunately, it seems that there is no convincing solution to these problems. The current situation is not satisfactory for another reason, too: It is almost common practice that artefacts (models, modelling languages etc.) are presented at conferences without being thoroughly discussed and evaluated by members of the corresponding scientific community. In our opinion, there is only one chance to overcome this problem (although we are not very optimistic): We have to put more emphasis on rational discourses that aim at a common evaluation - and comparison - of competing designs. In principle, those discourses have to be open for everybody who will live and work in the worlds to be constructed.

6 Conclusions

The prototypical approaches we have discussed so far confront us with a peculiar situation: The more promising the strategy to accomplish higher levels of automation at affordable cost, the bigger the epistemological challenges we have to face. One may argue that a constructivist approach should not be considered as a subject of scientific research - similar to the argument that engineering cannot be measured by the same criteria as science (for a discussion of this question with respect to Kuhn's theory of science see Wernick & Winder 1997). However, that would hardly solve the problem that there is lack of suitable criteria to evaluate and compare alternative artefacts to be used within information systems. It is no option either to neglect the construction of reference artefacts. Those artefacts are created and propagated anyway - by powerful vendors and standardisation organisations. They are an essential subject of the information systems discipline. Therefore, it seems a necessity that this discipline takes care of their evaluation. In fact, we would like to argue that the constructivist approach together with the establishment of procedures to evaluate the corresponding artefacts constitute the core of the information systems discipline. This would be similar to Weber (1997, pp. 69) who suggests that the core of the discipline should be the evaluation of information models/systems - except for the fact that we emphasize the constructivist nature of those

models.

Regarding the constructivist approach as a serious research strategy has substantial implications for the way information systems research is organised. First, it requires cross-disciplinary research, since it recommends to take into account aspects like human perception (of language and information artefacts), the psychology of organising, language as a design instrument (software engineering) and as a medium for social interaction and human thought. Second, the attempt to establish generic artefacts that could serve as references for a wide range of domains requires an amount of resources that can hardly be covered by a single institution. That implies the need to bundle research efforts and - at the same time - to improve the chances for a fruitful competition of alternative options. Against the background of the (social) reality of universities, these demands may seem naive. However, they are the only chance to oppose the large amount of redundancy caused by isolated research. Also, they offer the only chance for researchers at universities to maintain their profile against the growing dominance of the information systems industry and its subtle influence on research and teaching.

References

- Albert, H. (1968), *Traktat über kritische Vernunft*. Mohr, Tübingen
- Appelt, W. (1991), *Document Architecture in Open Systems. The ODA Standard*. Berlin, Springer, Heidelberg et al.
- Banville, C.; Landry, M. (1992), Can the Field of MIS be Disciplined? In: Galliers, R. (Eds.): *Information Systems Research. Issues Methods and Practical Guidelines*. Blackwell, London et al., pp. 61-88
- Couger, J.D. (1994), Measurement of the Climate for Creativity in I.S. Organizations. In: Nunamaker, J.F.; Sprague, R.H. (Eds.): *Proceedings of the 27th Annual Hawaii International Conference on System Sciences. Vol. IV*, IEEE Computer Society Press, Los Alamitos, Ca., pp. 351-357
- ESPRIT Consortium AMICE (1991), *Open system architecture: CIM-OSA AD 1.0*. Brussels 1991
- Frank, U. (1997), *Enriching Object-Oriented Methods with Domain Specific Knowledge: Outline of a Method for Enterprise Modelling*. Arbeitsberichte des Instituts für Wirtschaftsinformatik. Nr. 4, Koblenz
- Frank, U. (1998), *The MEMO Object Modelling Language (MEMO-OML)*. Arbeitsberichte des Instituts für Wirtschaftsinformatik. Nr. 9, Koblenz 1998
- Goldfarb, C.; Rubinsky, Y. (1990), *The SGML Handbook*. Clarendon Press, Oxford
- Habermas, J. (1981), *Theorie des kommunikativen Handelns. Vol. 1: Handlungsrationalität und gesellschaftliche Rationalisierung*. Suhrkamp, Frankfurt/M.
- Hirschheim, R. (1992), Information Systems Epistemology: An Historical Perspective. In: Galliers, R. (Ed.): *Information Systems Research. Issues Methods and Practical Guidelines*. Blackwell, London et al., pp. 28-60
- Hogart, R.M. (1980), *Judgement and Choice: The Psychology of Decision*. New York
- Kahneman, D.; Slovic, P.; Tversky, A. (1982) (Eds.), *Judgement under Uncertainty. Heuristics and Biases*. Cambridge
- Katz, R.L. (1990), Business/enterprise modelling. In: *IBM Systems Journal, Vol. 29, No. 4*, pp. 509-525
- Krogstie, J.; Lindland, O.I.; Sindre, G. (1995), *Towards a Deeper Understanding of Quality in Requirements Engineering*. In: Iivari, K.; Lyytinen, K.; Rossi, M. (Eds.): *Proceedings of the 7th Conference on Advanced Information Systems Engineering (CaiSE '95)*. Springer, Berlin, Heidelberg et al., pp. 82-95

- Lindland, O.I.; Sindre, G.; Sølvsberg, A. (1995), Understanding the Quality in Conceptual Modeling. In: IEEE Software, vol. 11, no. 2, pp. 82-95
- Lorenz, K. (1996), Sprache. In: Enzyklopädie Philosophie und Wissenschaftstheorie. Ed. by J. Mittelstraß. Vol. 4, Metzler, Stuttgart, Weimar, pp. 49-53
- Lorenzen, P.; Schwemmer, O. (1975): Konstruktive Logik, Ethik und Wissenschaftstheorie. Mannheim, Wien, Zürich
- Luhmann, N. (1984), Soziale Systeme. Grundriß einer allgemeinen Theorie. Frankfurt/M.: Suhrkamp, Frankfurt/M.
- Luhmann, N. (1967), Soziologische Aufklärung. In: Soziale Welt, Vol. 18, pp. 97-123
- Nilsson, N.J. (1986), Probabilistic Logic. In: Artificial Intelligence, vol. 28, pp. 71-87
- Object Management Group (1996), Common Facilities RFP-4: Common Business Objects and Business Object Facility. OMG, TC 13CF/96-01-04, (<http://www.omg.org/library/public-doclist.html>)
- The OpenDoc Design Team (1994), OpenDoc Technical Summary. Component Integration Laboratories (<http://www.cilabs.org/>)
- Perrow, C. (1986), Complex Organizations: A Critical Essay. 3rd ed., Glenview, Ill.
- Petrie, C.J. (1992) (Ed.), Proceedings of the First International Conference on Enterprise Integration Modeling. MIT Press, Cambridge, Mass.
- Pfeffer, J. (1978), The Ambiguity of Leadership. In: McCall, M.W.; Lombardo, M.M. (Eds.): Leadership: Where else can we go? Durham, NC 1978, pp. 13-34
- Popper, K.R. (1935): Logik der Forschung. Wien 1935
- Ragowsky, A.; Ahituv, N.; Neumann, S. (1996): Identifying the value and importance of an information system application. In: Information & Management, No. 2, Vol. 31, pp. 89-102
- Saarinen, T. (1996), An expanded instrument for evaluating information system success. In: Information & Management, No. 2, Vol. 31, pp. 103-118
- Savage, C.M. (1990). Fifth generation management - integrating enterprises through human networking. Digital Press
- Schank, R.C. (1985), The Cognitive Computer. On Language, Learning and Artificial Intelligence. Addison-Wesley, Reading, Mass.
- Simon, H.A. (1964) The Corporation: Will It be Managed by Machines. In: Leavitt, H.J.; Pondy, L.R. (Eds.): Readings in Managerial Psychology. Chicago, London, pp. 592-617
- Simon, H.A. (1983) Search and Reasoning in Problem Solving. In: Artificial Intelligence, vol. 4, pp. 181-201
- Simons, H.W. (1976), Persuasion. Addison-Wesley, Reading, Mass. 1976
- Sipior, J.C.; Ward, B.T. (1997), Sexual Harassment via Employee E-mail: An International Issue or just a United States Quirk? In: Galliers, R. et al. (Eds.): Proceedings of the 5th European Conference on Information Systems. Cork, Vol. I, pp. 313 - 320
- Slovic, P.; Fischhoff, B.; Lichtenstein, S. (1977), Behavioral Decision Theory. In: Annual Review of Psychology, Vol. 28, pp. 1-39
- Sowa, J.F.; Zachman, J.A. (1992), Extending and formalizing the framework for information systems architecture. In: IBM Systems Journal, Vol. 31, No. 3, pp. 590-616
- Van der Spek, R.; De Hoog, R. (1995), A framework for a knowledge management methodology. In: Wiig, K. (Ed.): Knowledge Management Methods. Schema Press, Arlington/Texas 1995, pp. 379-393
- Walsh, J.P.; Ungson, G.R. (1991), Organizational Memory. The Academy of Management Review, No. 1, Vol. 16, pp. 57-91
- Weber, R. (1997), Ontological Foundations of Information Systems. Coopers&Lybrand, Melbourne
- Weick, K.E. (1980), The Social Psychology of Organizing. 2nd ed., Addison-Wesley, Reading, Mass.

Wernick, P.; Winder, R.L. (1997), Software Engineering as a Kuhnian Discipline. In: Winder, R.L.; Probert, S.K.; Beeson, I.A. (Eds.): Philosophical Aspects of Information Systems. Taylor&Francis, London, pp. 117-129

Wiig, K.: Knowledge Management. The Central Management Focus for Intelligent-Acting Organizations. Vol. 2., Schema Press, Arlington/Texas

Wittgenstein, L. (1971), Philosophische Untersuchungen. Suhrkamp, Frankfurt/M.

Wittgenstein, L. (1981), Tractatus logico-philosophicus. Routledge & Kegan Paul

Zadeh, L.A. (1975), Fuzzy Logic and Approximate Reasoning. In: Synthese, Vol. 30, pp. 407-428