2009

Overcoming Language Barriers for Business Process and Information Systems Integration – A Theoretical Model

Marcus Laumann
Johann Wolfgang Goethe Universitat Frankfurt am Main, marcus.laumann@bertelsmann.de

Christoph Rosenkranz
Johann Wolfgang Goethe Universitat Frankfurt am Main, rosenkranz@wiwi.uni-frankfurt.de

Overcoming Language Barriers for Business Process and Information Systems Integration – A Theoretical Model

Marcus Laumann
Goethe University, Germany
Marcus.Laumann@bertelsmann.de

Christoph Rosenkranz
Goethe University, Germany
Rosenkranz@wiwi.uni-frankfurt.de

ABSTRACT

"The limits of my language are the limits of my mind. All I know is what I have words for." (Wittgenstein)

Supply chain management (SCM) deals with the management of flows of goods, funds and information within and between supply chain partners in order to satisfy customer needs in the most efficient way. Consequently, business processes and information systems (IS) must be integrated between supply chain partners. However, many business process and information system integration projects fail, are postponed or exceed budget. Following the outcome of different explorative field studies, we elaborate on a theoretical explanation for this phenomenon based on philosophy of language. By characterizing IS as language communities, we highlight the importance of using domain-specific languages for the success of these projects and introduce a theoretical cost model for analyzing business process and IS integration projects.

Keywords

Business process integration, information system integration, system design, philosophy of language.

INTRODUCTION

SCM aims at optimizing the flow of goods, funds and information within and between supply chain partners from a global perspective (Christopher 1998). Any organization needs effective information flows to coordinate and control its activities (e. g., Galbraith 1977; March and Simon 1958; Tushman and Nadler 1978). Hence, the proper integration of business processes and IS is becoming more and more important. Empirical surveys indicate high costs and high overall failure rates of business process and IS projects (e. g., BCS 2004; SGI 2001). Also, many projects are postponed or exceed budget. The reasons are manifold and include, amongst others, the necessity for coordination between the various stakeholders involved (e. g., Gallivan and Keil 2003; Joshi et al. 2007; Ko et al. 2005). Therefore successful communication between involved stakeholders is deemed to be one of the main drivers for project success (e. g., Gallivan and Keil 2003; Ribbers and Schoo 2002; Vlaar et al. 2008).

Existing IS research usually distinguishes between three IS development phases (e. g., Hirschheim et al. 1995; Scheer 1999): (1) requirements specification, (2) design specification and (3) implementation. The exact definition of requirements is especially crucial for the overall success of any IS integration project (e. g., Davis 1990). We argue that one of the main reasons for problems and issues in IS integration projects lies in the ambiguity and vagueness of natural language (e. g., Al-Rawas and Easterbrook 1996; Berry and Kamsties 2003 in the context of IS projects; see Eco 1984; Radman 1997 in general). Ambiguity in this context means the existence of multiple and conflicting interpretations about an organizational situation and gives a measure of the organizational agents' ignorance of whether a variable exists (Daft and Lengel 1986, p. 556; Daft and Macintosh 1981; Weick 1979, pp. 4-7). For example, as companies develop specialized units, each of these units generates its own idiosyncratic norms, values, time frame, terminologies and coding schemes to permit effective information processing (Tushman 1979, p. 590). Also different or competing terms might be used for the same thing. This becomes extremely relevant for SCM where people from different organizations have to work and communicate with each other. Communication within supply chain projects dealing with business process and IS integration is especially complex due the high number of participants from different organizations and interactions. Not only will different organizations have a mismatch in terminologies and coding schemes, but differentiated subunits within the same organization will likely also have contrasting terminologies and coding schemes. For example, specific terms associated with a newly developed application system, spanning the entire supply chain and having an effect on every supply chain partner, are interpreted differently by different people, or the same function of the application systems is described by different people using not quite the same words. Consequently, ambiguity in such projects is primarily due to language-based interpretation, communication, and sense-making: the unique human mode of word meaning and symbolic reference (Deacon 1997, p. 43). In organization theory the research is rife with evidence (Mintzberg 1973) that organizational life is characterized by a substantial amount of communication: in meetings, conferences, and social events that fill the everyday life of workers and managers. The meaning of words corresponds to the observable behavior that people using those words produce in each other's presence; this meaning is a dialect particular to those people. Natural language forms the basis for each interactions and is enhanced the as people work and act together in a certain domain. Consequently, cultural and community-related differences can be directly observed in language differences (Weber and Camerer 2003, p. 404). Such dialects or *domain-specific languages* (DSLs) carry significant economic value for their users, because they make redundant the iterative specification of all the words that embody them, since dialects are common knowledge among their participants (Moldoveanu 2002, p. 239).

Furthermore, diagrams and conceptual models can be built by using *domain-specific modeling languages* (i. e., more formal variants of DSLs) and are intensively used in this context by software engineers, system analysts or business process engineers for specifying requirements of new or changing applications and business processes (Davies et al. 2006; Moody 2005). Wand and Weber (2002) claim that such conceptual models are designed for at least three different purposes: (1) *to support communication between developers and users*, (2) *to help analysts understand a domain*, and (3) *to provide an input to system design*. For example, the *model-driven architecture* (MDA) literature proposes that models using DSLs can be employed for automatically configuring application systems for specific domains (e. g., Dreiling et al. 2006). Nonetheless, MDA still relies on human modelers with domain-specific knowledge and sufficiently familiar with the context-specific DSLs. While the vision of MDA can today be realized in narrow contexts and uncomplicated situations, automatically building specific application systems for complex, domain-specific situations is still a far way off and mostly relies on human communication.

Figure 1 shows different participants and interactions while integrating business processes and IS. The scenario is based on the description and outcome of previous field studies which are similar to this situation (e. g., Laumann 2008; Laumann and Rosenkranz 2008; Rosenkranz et al. 2008; Rosenkranz et al. 2009).In one of the field studies a requirements specification for a company spanning Supply Chain Controlling System (Laumann 2008) was carried out that involved the integration of business and IT employees of different organizations. In such a scenario, many so-called "language barriers" have to be overcome, since people from different organizations have to communicate with each other, and business experts (e. g., logistics, finance, transport, quality, etc.) and ICT professionals have to understand the exact meaning of the requirements for integrating businesses. The complexity is very high in a supply chain because not only do employees with different backgrounds and from context-dependent situations have to negotiate a shared problem understanding, but even company-spanning communication and negotiation is required. Moreover, the risk of misunderstanding due to misinterpretation of information is always present while people communicate in a non-standardized way; natural language is vague, ambiguous and error-prone.

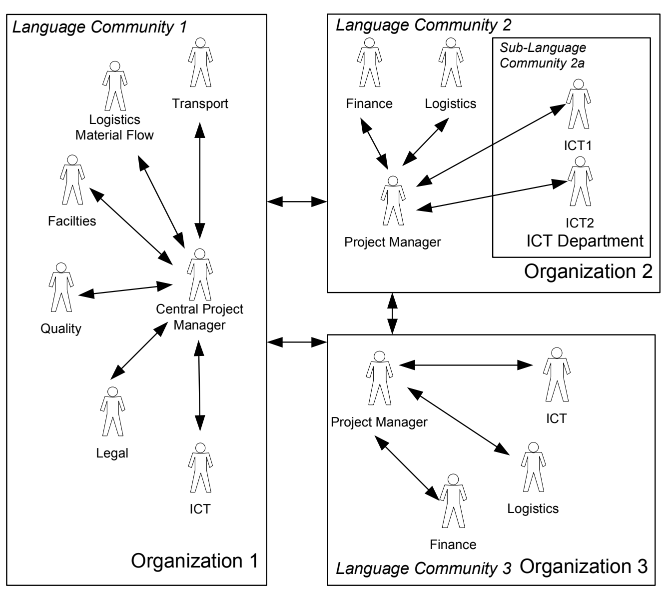

Figure 1. Communication in project and language communities

In this paper we will only address what we call “unconscious ambiguity”, and what Clark (1996) denotes as “illusion of evidence”, and exclude any intentional or deliberate ambiguity, e. g., due to politicking or other motives. Figure 1 sketches the different language barriers that can be found in such a project. This highlights our viewpoint that organizations can be interpreted as different *language communities* (Kamlah and Lorenzen 1984), (Holten 2007), and even within organizations, every expert might belong to a different sub-language group (e. g., ICT department or logistics unit) and uses certain terms that might not be known or interpreted differently by other experts.

Communication takes place between the different project participants. Usually a project manager in each organization coordinates the activities of the corresponding project partner internally in order to simplify and filter the communication. Only relevant information is forwarded to the local experts. Project managers communicate with each other across organizations and coordinate the different tasks internally within each organization. Therefore we argue that the project managers must be able to “understand” and “speak” the languages of the other organizations involved and also internally translate requirements and feedback into the sub-languages spoken in the different departments. Communication in and between organizational units is taking place in jargon and technical, DSL. In addition, the overall communication time of any such project is highly influenced by how effective this communication is organized. If everybody talks to everybody, the communication time is very high. As highlighted before one task of management is to structure communication, therefore the project managers have to coordinate the internal and external communication properly and should also decide when to introduce a DSL. In addition, while communicating with each other, misunderstandings between the different sub-unit groups have to be avoided in order to reduce costly communication and reworking time.

Effective communication is critical to the development of mutual understanding in systems design, however, the process by which this mutual understanding is achieved is not well understood (Tan 1994). Almost every research argues that to facilitate communication between stakeholders is a necessity, and it is generally assumed that graphical presentation improves comprehension; however, it has been shown that graphic displays are not always easily understood and require learning (Nordbotten and Crosby 1999). In this paper, we do not join the ranks of either the proponents or the opponents of these perspectives. Rather, and more importantly, we point out that none of the existing approaches – be it cognitive (e. g., Evermann 2005; Gemino and Wand 2003), ontological (e. g., Wand and Weber 1993; Wand and Weber 1995), or linguistic (e. g., Ågerfalk and Eriksson 2004; Auramäki et al. 1988; Goldkuhl and Lyytinen 1982; Lyytinen 1985) – can until now theoretically explain what *effect* DSLs really have on the IS development process in complex IS integration projects. Based on the presented business integration example, we elaborate on a theoretical cost model in order to provide a decision basis when the introduction of DSL in similar project situations makes sense to reduce overall probability of misunderstandings and thereby total project costs. We aim at providing guidelines if and when to introduce a DSL in order to reduce the overall rate of failed and postponed business process and IS integration projects. According to Fortune and White (2006), “good communication” is one of the most claimed critical success factors in complex projects. Furthermore, project cost budgeting can be eased by providing and using such a cost model, in order to improve budget estimations. A decision base is required in order to determine if and when the introduction of a DSL is advantageous from a cost perspective.

For doing this, we at first introduce our theoretical standpoint: We interpret organizations as language communities and show that especially in complex projects ambiguity of language is critical for overall project success and costs. Then, we elaborate on a concept based on philosophy of language for explaining the mentioned issues from a theoretical perspective. Then we introduce a theoretical cost model in order to analyze total project costs with and without using a DSL when integrating businesses and information systems within complex settings. Finally, we summarize our findings and give an outlook to future research.

THEORETICAL STANCE – FORMING LANGUAGE COMMUNITIES

In this paper we interpret organizations as language communities that are formed by the communication taking place between all participants. If members of a group of people communicate and each has an aligned semantic and pragmatic dimension of a symbol (or term) in mind then this group of people forms a language community (Kamlah and Lorenzen 1984). From a theoretical stance, each organization in a supply chain forms a separate language community which establishes a “common ground” of understanding (e. g., Clark 1992; Clark 1996; Kamlah and Lorenzen 1967; Kamlah and Lorenzen 1984). Sub-languages in organizations and especially in supply chains (as networks of organizations) exist and must be bridged. A consensus of meaning in defining and interpreting requirements has to be ensured when integrating business processes and IS, and participants have to agree on a common meaning system. However, it is known that information processing is prone to errors while it is being transferred from one person to another (e. g., Boisot and Canals 2004; Boisot and Li 2006; Deacon 2007; Langefors 1995). A person who receives information might interpret it as it was not exactly meant by the person who gave it to her or him because she or he does not the coding scheme or does not understand the used jargon. These decoding errors lead to deviations between the sent and received message and can result in unintentional design of business processes and IS. Therefore linguistic communication is of paramount importance for IS research (Holten 2007; Lyytinen 1985; Winograd 1988; Winograd 2006). Furthermore, recent studies suggest that more attention should be given to the social act of adaptation of IS by organizational agents (Avgerou et al. 2004; Vaast and Walsham 2005). This implies a focus on social processes, including issues such as agents' meaning constructions and sense-making. The resulting understanding recognizes that it becomes increasingly important to study the meanings that organizational agents ascribe to IS, given the local context in which they are to use IT, and in which their meanings about IT are constructed. This includes language and communication.

Backlund argues that an organization is complex when "(1) it consists of many components or subsystems ("parts"), (2) and/or there are many relations and/or interactions between the components and subsystems, (3) and/or when these relations are not symmetric, (4) and/or when the arrangement of the components and/or subsystems is not symmetric" (Backlund 2002, p. 33). Following this definition, the complexity of supply chain projects is especially high because the number of participants and interactions (communication) is extremely high and this communication needs to be coordinated if it should not be assymetric. The complexity within a certain project domain directly influences the probability of misunderstandings. Ambiguity in natural language is the crucial problem when overcoming language barriers within organizations. Every perception of the world is language bound and language is the mediator between reality and each individual. Communication processes are often defective or pathological (Espejo et al. 1996, p. 28). Every language transfers not an objective message; it also contains a certain perspective on and interpretation of the real world. It is strongly necessary that sender and receiver have a minimum set of jointly defined vocabulary and understanding in order to communicate effectively while integrating businesses. Conventions are a precondition to achieve a shared understanding. Furthermore, coding and decoding should be as easy as possible to avoid misunderstandings. The risk of confusion arises when the sense and meaning of language is translated into another language.

Following Ashby's Law of Requisite, "Only variety can destroy variety" (Ashby 1964, p. 207), we argue that the variety exhibited by the different language communities (experts, departments within organizations and organizations along a supply chain) must be matched by a similarly high variety. This can basically be achieved by having mediators between the different language community barriers. A mediator might be a participant who understands and speaks the languages of the language communities involved or a DSL; in other words, a boundary spanner (e. g., Gasson 2006; Pawlowski et al. 2000; Tushman 1977). By introducing a new DSL for all project members, the number of states of the domain in scope can also be reduced and ambiguity of natural language is reduced to a minimum by reducing the number of possible states of a certain problem domain. The DSL is thereby a filter. An inter-subjective understanding is achieved.

A new DSL is learned by practical learning of meaning by "acting" together (Kamlah and Lorenzen 1984). Agreement must be achieved about definition *and* judgement. A grammar, rules and norms for meaningful language and judgement are required. The form of life (culture, context, history) enables language to function (even in supply chain scenarios). Language constructs need to be introduced, discussed and jointly explained. The meaning of a word is its use in language: one understands a word when one knows how to use it. To create a joint understanding and to align the meaning of terms, it is required to act together. We argue that only face-to-face meetings have enough richness and variety (Daft and Lengel 1986) to guarantee that a joint understanding is created by acting together because empractical learning means to experience what the meaning of a term is in concrete situations. If there is confirmation for a certain interpretation in different situations, then meaning is formed in the perceived reality of the language community. By repeating the meaning a new "term" is formed. By using the DSL within the language community the perceived reality can be checked and improved. Hence, participants of the language community must adjust their perception of reality and achieve a common understanding. The introduction of a DSL comes therefore at certain costs (e. g., Nikolopoulos and Holten 2007). By the introduction of a DSL the complexity of a phenomenon can be reduced and communication is eased and becomes more efficient. A language community with a terminology with a lower number of terms (reduced vocabulary) than colloquial language ensures more efficient speech, actions and decision making. Figure 2 shows the process of creating a DSL to ease communication between four project participants from two different organizations. Each project member perceives a certain setting, interprets it and constructs a certain reality of a project domain, before discussing it with the other project members. The modeling of processes or IS is from our standpoint the introduction of a DSL. Modeling languages thereby avoid disruptions of communication by excluding any accentuation, volume, gesture of natural language, since they formalize the perception of reality in an abstract from, thereby reducing the bias of interpretations. By introducing a DSL different perceptions of reality are integrated into an agreed model of reality. The aim is to create one common reality about a situation within a certain domain of interest. Communication should therefore be standardized, especially in complex situations, to ease communication and reduce overall communication and reworking costs. Looking at a typical modeling process for business and IS integration one can distinguish between different levels of a model (Stachowiak 1973).

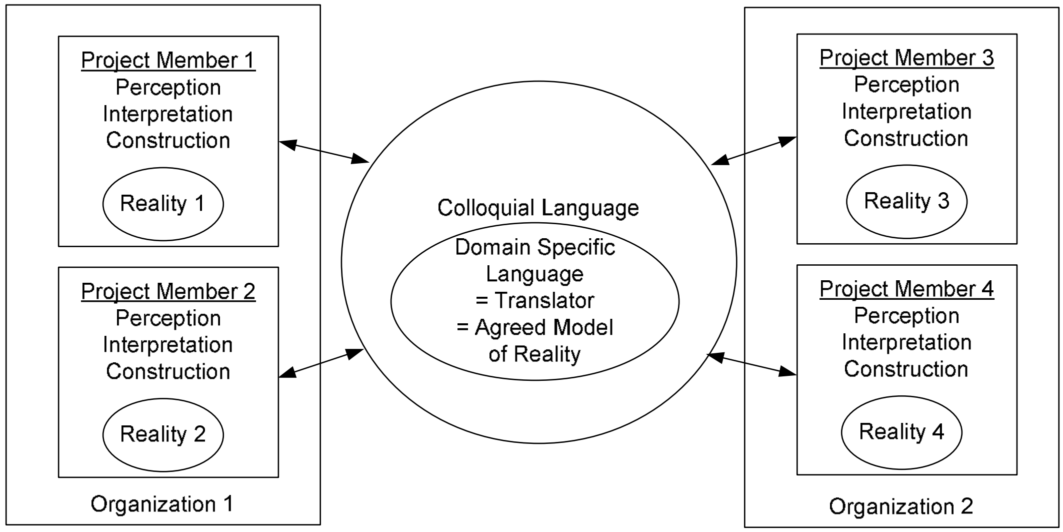

Figure 2. Act of creating a domain specific language to form a language community

Figure 3 shows how an object domain (e.g. a certain business process or IS integration setting) is perceived by two project members that have to model the same object domain. The perceived model is interpreted and based on this internal model; then an external model is created. All of these models may differ. However, both project members have to align the constructed external models. By jointly discussing and agreeing on a model and its meaning with the help of colloquial language, a language community is formed and consensus is achieved.

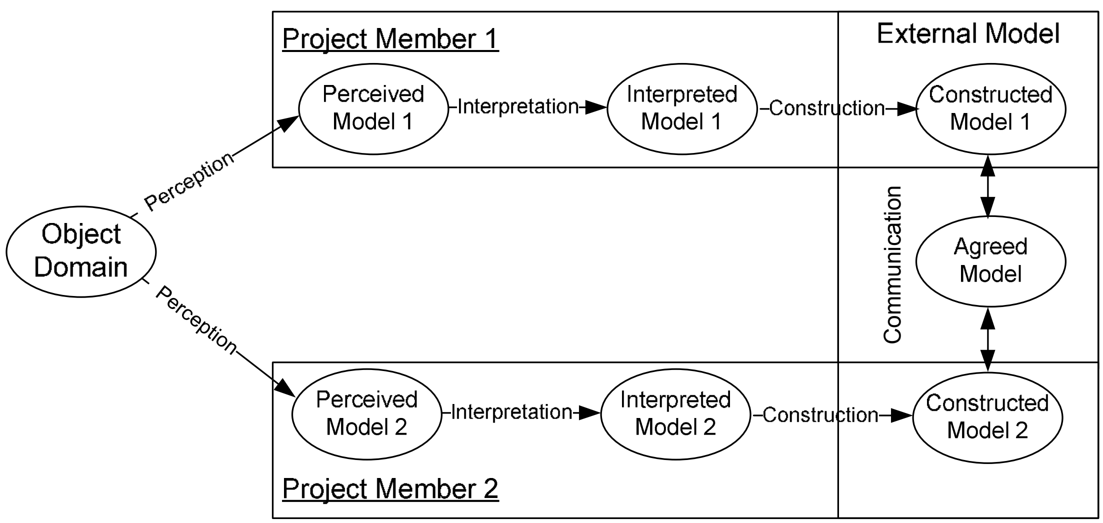

Figure 3. Act of creating a model to form a language community (based on Stachowiak's (1973) levels of a model)

In the following we will assume that all project participants have the skills to fulfil their specific task in a business process and IS integration project. The proper perception and interpretation of their object domain is therefore given. The perception, interpretation and construction of reality is the invention of each individual. However, the constructed models have to be aligned. This is the scope of this paper. By discussing the different constructed models (representing different perceptions, interpretations and constructions of reality), agreeing on meanings and a joint model, an inter-subjective understanding of the problem domain is created and encapsulated in the new model. By interacting with other participants, it is possible for each project member to check the viability of her or his internal models and to confirm or adjust them. In different field studies (Rosenkranz et al. 2009; Laumann 2008; Laumann and Rosenkranz 2008) the introduction of a DSL, especially designed for the modeling of information flows within organizations, helped to improve the quality and efficiency of communication of all project participants while integrating businesses. The formalized approach helped to quicker reach consensus about requirements and helped to translate business requirements into technical requirements. Basically, we claim that three different effects exist that have to be evaluated when introducing a DSL (see also Nikolopoulos and Holten 2007 for a related argument):

1) *There are setup costs when creating language communities*. These costs are related to the complexity of the domain and the number of people involved.

2) *A domain-specific language makes communication more efficient*. Language communities are formed faster when introducing a DSL. This is especially valid in complex problem domains.

3) *The introduction of a domain-specific language improves the quality of communication and thereby reduces the probability of misunderstandings*. Reduced misunderstandings lead to reduced reworking costs in later phases of the project.

Summing up, agreeing on a model may provide consensus about requirements and reduces complexity by simplifying the object domain and just focussing on the points that are essential for differentiation within the object domain. The risk of ambiguity of natural language is reduced since the probability of misunderstandings is reduced. This leads to less overall communication costs and less rework in later business integration phases. Moreover, communication is faster since specific well-known terms are used that do not have to be explained over and over again. We now elaborate on a cost model to highlight the mentioned cost effects.

PROJECT COSTS FOR BUSINESS PROCESS AND INFORMATION SYSTEM INTEGRATION

As discussed before, the introduction of a DSL (e. g., a model) comes at certain cost. The introduction makes sense only if reduction in communication and re-working costs compensate these language construction costs. We therefore elaborate on a cost model in order to provide a first decision basis if and when the introduction of a DSL makes sense. A general cost model is provided in order to test our hypothesis in quantitative studies in the future. We elaborate on a mathematical model to evaluate the effect on communication and reworking costs when introducing a DSL (a modelling technique) that reduces the probability of misunderstandings for a specific domain. Let W be the working time of an employee for a certain task. The working time for a certain task is then defined as the sum of the original production working time (PWO), the reworking time (PWR) and communication time (CW) for this task (adopted from Jin & Levitt (1996), p. 175).

$$W = PWO + PWR + CW$$

The total working time (TW) for a certain task m is defined as the sum of the working time of all employees involved in the specific task. In our case introduced before (compare figure 1) the task is the integration of a business process and IS in a supply chain.

$$TW_m = \sum_{i=1}^{n} PWO_{mi} + PWR_{mi} + CW_{mi}$$

i : employee i with i >= 1 $\in$ N, n = no. of employees involved in task m

PWO_{mi} = original production working time, PWR_{mi} = reworking time, CW_{mi} = communication time

In order to minimize the costs involved in fulfilling the business and IS integration task m, it is necessary to minimize the total working time. We assume that the original production time is fixed and cannot be reduced for a given set of employees (compare assumptions on next page). Hence, the focus is on minimizing the reworking production time and the communication time for all project members:

$$\sum_{i=1}^{n} PWR_{mi} + CW_{mi} = \min$$

For our model we will assume in the following:

1. All employees involved have the required skills and knowledge to fulfil their tasks and are pre-defined. PWO is therefore assumed to be fixed.

2. All employees involved want to fulfil their tasks and there are no political reasons for the rework produced (no intentional ambiguity or forwarding of wrong or obsolete information).
3. The only explanation for PWR are misinterpretations between sender and receiver; if all misunderstandings can be avoided this means PWR = 0 for all employees involved.

In the following three effects are considered:

1. Let l (language construction costs) be total time spent for introducing a DSL to all employees.
2. Let $\sum_{i=1}^{n} \Delta CW_{mi}$ be the reduction in communication costs after having introduced a DSL (communication costs when using a DSL - communication costs without using DSL) and $\Delta CW_{mi} < 0$.
3. Let $\sum_{i=1}^{n} \Delta PWR_{mi}$ be the reduction in re-working costs when using a DSL (reworking costs when using a DSL - reworking costs without using DSL) and ΔPWR_{mi} = reduced reworking time, $\Delta PWR_{mi} < 0$.

Let M be a modelling technique that explains a certain object setting S and reduces the probability of misunderstandings, P(M). The introduction of a modelling technique M that avoids misunderstanding makes sense, if the expected costs are reduced when using the modelling technique (M). The expected costs (E(X)) for a specific task are defined as follows:

$$E(X) = \sum_{j=1}^{m} x_j p_j$$

with: x_j = costs (measured in time spent), p_j = probability that X = x_j

The expected cost effect when using a modelling technique is:

$$= l + (\sum_{i=1}^{n} \Delta PWR_i) * P(M) * P(A \mid M) + 0 * P(M) * P(\overline{A} \mid M) + 0 * P(\overline{M}) + \sum_{i=1}^{n} \Delta CW_i$$

$$= l + (\sum_{i=1}^{n} \Delta PWR_i) * P(M) * P(A \mid M) + \sum_{i=1}^{n} \Delta CW_i$$

With:

l = total time spent for introducing modelling technique M (language construction costs) for employee i,

$P(M)$ = probability of misunderstanding

$P(\overline{A} \mid M)$ = probability that modelling technique M does not avoid misunderstanding when a misunderstanding occurs (conditional probability),

$P(A \mid M)$ = probability that modelling technique M avoids a misunderstanding when a misunderstanding occurred (conditional probability)

Explanation of the terms of the sum mentioned above:

1. the modelling technique will be introduced, even if there is no misunderstanding later on;
2. rework is reduced, if there is a misunderstanding and the modelling technique avoids this misunderstanding;
3. the introduction of the technique and rework is not reduced (=0), if a misunderstanding occurs and the modelling technique doesn't avoid this misunderstanding;
4. no rework is required if there is no misunderstanding;
5. the communication is eased by the DSL and independent of the probability of misunderstandings.

The complexity of the problem domain is influencing $P(M)$ and $P(A \mid M)$ can be interpreted to be a quality measure for the introduced DSL. The introduction of a DSL is only advantageous, if:

$$l + (\sum_{i=1}^{n} \Delta PWR_i) * P(M) * P(A \mid M) + \sum_{i=1}^{n} \Delta CW_i < 0$$

⇔

$$l < -(\sum_{i=1}^{n} \Delta PWR_i) * P(M) * P(A \mid M) - \sum_{i=1}^{n} \Delta CW_i$$

If the construction costs are less than the sum of the reduction of rework (due to reduction of misunderstandings) and communication costs, then the introduction of a DSL makes sense. Figure 4 sketches the construction costs for setting up a new DSL and compares the communication and reworking costs when using colloquial and DSL in different complex problem domains. The link between the graphs and the introduced formulas is also highlighted.

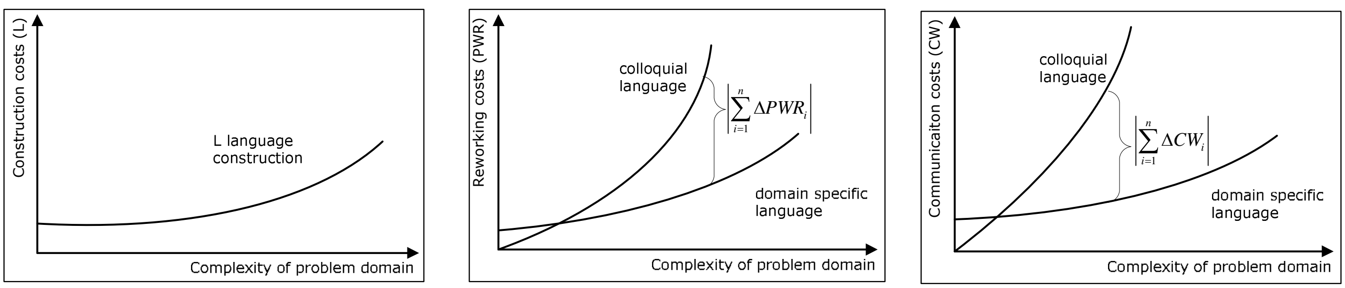

Figure 4. Construction, communication and reworking costs when using colloquial and domain-specific language, adopted from Nikolopoulos and Holten (2007)

Now we highlight the domain-specific construction costs, the communication and reworking costs by consolidating them into one total cost graph (Figure 5).

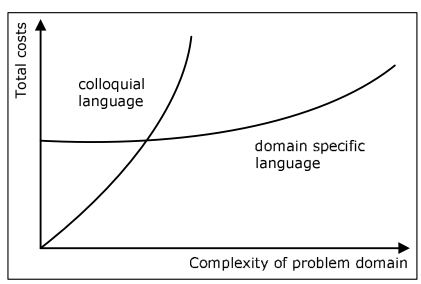

Figure 5. Total costs in different complex situations, adopted from Nikolopoulos and Holten (2007)

For a typical business integration scenario (e.g. the implementation of a Supply Chain Controlling System) with a suitable DSL (compare Laumann 2008) we assume the following variables:

n = 10 employees were involved, 4 business people and 6 IT specialists

$l = 1h$ (the introduction of the DSL (modelling technique) took 1 hour)

$P(M) = 0,2$ (the probability of misunderstandings equals 20 percent)

$P(A \mid M) = 0,9$ (in 90% of all cases the misunderstanding is avoided due to the DSL)

$$\sum_{i=1}^{n} \Delta PWR_i = n*(-2h) = 10*(-2h) = -20h$$

(each employee saves on average 2 hours of reworking time in case of an avoided misunderstanding)

$$\sum_{i=1}^{n} \Delta CW_i = n*-0,25 = 10*-0,25h = 2,5h$$

(each employee saves on average 15 minutes of communication time by using the DSL)

Applying these variables to our model leads to:

$$l < -(\sum_{i=1}^{n} \Delta PWR_i)*P(M)*P(A \mid M) - \sum_{i=1}^{n} \Delta CW_i$$

1h < 20h*0,9*0,2 + 2,5h

1h < 3,6h + 2,5h = 6,1h

Hence, the DLS introduction costs are lower than the sum of the reduced communication and reworking costs. In this specific case the introduction of the DSL would be advantageous.

Summing up, the introduction of a domain-specific-language can lead to reduced total project costs by reducing communication and reworking costs in complex settings. The introduction of a DSL makes no sense in non-complex situations, because language construction costs are expected to be higher than the reduced reworking and communication costs. As mentioned before, supply chain projects are extremely complex due to the number of different participants from different language communities. Therefore we argue that the introduction of a DSL reduces overall costs in supply chain scenarios. Successful and good business integration projects are able to form language communities quickly and to use DSL as a language mediator for project participants. Furthermore, our model suggests why it is that some business process and IS integration projects fail, because no DSL is introduced and colloquial language cannot be used to explain complex situations.

CONCLUSION AND OUTLOOK

The set-up of a company spanning IS and business processes is very complex, because different sub-languages are spoken in organizations. Misunderstandings between participants occur in daily business, are crucial and must be avoided while integrating businesses. The introduction of DSL can help to reduce the probability of misunderstandings and reworking costs. Moreover, communication in complex situations is eased and costs are reduced in complex situations. However, the introduction of DSL comes at certain construction costs which have to be considered. Participants must work out jointly the language and meanings by working and acting together and testing the meaning of words in practice. The DSL is the lowest common denominator for all participants and serves as a "translator" into the different languages of the project participant. Based on a first mathematical model we elaborated on providing a decision basis, if and when the introduction of a modeling technique (an instance of a DSL) makes sense from a cost perspective. The model can be applied by estimating the influencing factors. The language introduction and reduced communication costs and the conditional probability can be measure for a specific DSL, the reworking costs can be estimated by measuring the time required for rework due to misunderstandings, and the probability of misunderstandings can be measured for specific types of settings. Also the probability of misunderstandings with and without using a modeling technique must be empirically tested in different complex situations for a certain modeling technique. The interactions (communication) have to be considered more explicitly in our cost model; at this stage we assume the same communication structure with and without using a modeling technique. Furthermore, we need to define how, based on our definition and influencing variables, complexity can be assessed in a specific setting. Future research will address these limitations. The conditional probability $P(A \mid M)$ as a possible measure for the quality of a DSL needs also further development. We argue that the probability of misunderstandings is the important factor for measuring the goodness of language communities. In this paper we provided a theoretical model in order to explain how the introduction of a DSL helps to reduce overall project costs in complex settings. Future research will therefore focus on testing these propositions and on providing guidelines for quickly and successfully introducing DSL and reaching shared understanding.

REFERENCES

1. Ågerfalk, P.J., and Eriksson, O. "Action-oriented conceptual modelling," *European Journal of Information Systems* (13:1) 2004, pp 80-92.
2. Al-Rawas, A., and Easterbrook, S. "Communication problems in requirements engineering: a field study," First Westminster Conference on Professional Awareness in Software Engineering, London, UK, 1996.
3. Ashby, W.R. *An Introduction to Cybernetics*, London, 1964.
4. Auramäki, E., Lehtinen, E., and Lyytinen, K. "A speech-act-based office modeling approach," *ACM Transactions on Information Systems (TOIS)* (6:2) 1988, pp 126-152.
5. Avgerou, C., Ciborra, C., and Land, F.F. (eds.) *The Social Study of Information and Communications Technology: Innovation, Actors and Context*. Oxford University Press, Oxford, UK, 2004.
6. Backlund, A. "The concept of complexity in organisations and information systems," *Kybernetes* (31:1) 2002, pp 30-43.
7. BCS "The Challenges of Complex IT Projects," The British Computer Society (BCS) and The Royal Academy of Engineering, London, UK.
8. Berry, D.M., and Kamsties, E. "Ambiguity in Requirements Specification," in: *Perspectives on Software Requirements,* S.d.P. Leite and K. J. H. Doorn (eds.), Academic, 2003, pp. 191-194.
9. Boisot, M., and Canals, A. "Data, information and knowledge: have we got it right?," *Journal of Evolutionary Economics* (14:1) 2004, pp 43-67.
10. Boisot, M., and Li, Y. "Organizational versus Market Knowledge: From Concrete Embodiment to Abstract Repesentation," *Journal of Bioeconomics* (8) 2006, pp 219-251.
11. Christopher, M. *Logistics and Supply Chain Management. Strategies for Reducing Cost and Improving Service*, (2 ed.) Financial Times Professional, London, UK, 1998.
12. Clark, H.H. *Arenas of Language Use* Chicago University Press, Chicago, IL, USA, 1992.
13. Clark, H.H. *Using Language* Cambridge University Press, Melbourne, 1996.
14. Daft, R.L., and Lengel, R.H. "Organizational information requirements, media richness and structural design," *Management Science* (32:5) 1986, pp 554-571.
15. Daft, R.L., and Macintosh, N.B. "A Tentative Exploration into the Amount and Equivocality of Information Processing in Organizational Work Units," *Administrative Science Quarterly* (26:2) 1981, pp 207-224.
16. Davies, I., Green, P., Rosemann, M., Indulska, M., and Gallo, S. "How do Practitioners Use Conceptual Modeling in Practice?," *Data & Knowledge Engineering* (58:3) 2006, pp 358-380.
17. Davis, A.M. "The Analysis and Specification of Systems and Software Requirements," in: *Systems and Software Requirements Engineering,* R.H. Thayer and M. Dorfman (eds.), IEEE Computer Society Press-Tutorial, 1990, pp. 119-134.
18. Deacon, T.W. *The Symbolic Species. The Co-evolution of Language and the Brain* W. W. Norton & Company, New York, NY, USA, 1997.
19. Deacon, T.W. "Shannon – Boltzmann – Darwin: Redefining information (Part I)," *Cognitive Semiotics* (1:1) 2007, pp 123-148.
20. Dreiling, A., Rosemann, M., Aalst, W.v.d., Heuser, L., and Schulz, K. "Model-based software configuration: patterns and languages," *European Journal of Information Systems* (15:6) 2006, pp 583-600.
21. Eco, U. *Semiotics and the philosophy of language* Indiana University Press, Bloomington, IN, USA, 1984.
22. Espejo, R., Schuhmann, W., Schwaninger, M., and Bilello, U. *Organizational Transformation and Learning. A Cybernetic Approach to Management*, Chichester, UK et al., 1996.
23. Evermann, J. "Towards a coginitive foundation for knowledge representation," *Information Systems Journal* (15:2) 2005, pp 147-178.
24. Fortune, J., and White, D. "Framing of project critical success factors by a systems model," *International Journal of Project Management* (24:1) 2006, pp 53-65.
25. Galbraith, J.R. *Organization Design* Addison-Wesley, Reading, MA, USA, 1977.
26. Gallivan, M.J., and Keil, M. "The user-developer communication process: a critical case study," *Information Systems Journal* (13:1) 2003, pp 37-68.
27. Gasson, S. "A genealogical study of boundary-spanning IS design," *European Journal of Information Systems* (15:1) 2006, pp 26-41.
28. Gemino, A., and Wand, Y. "Evaluating Modeling Techniques based on Modles of Learning," *Communications of the ACM* (46:10) 2003, pp 79-84.
29. Goldkuhl, G., and Lyytinen, K. "A language action view of information systems," 3rd International Conference on Informations Systems (ICIS 1982), Houston, TX, USA, 1982.
30. Hirschheim, R., Klein, H., and Lyytinen, K. *Information Systems Development and Data Modeling. Conceptual and Philosophical Foundations* Cambridge University Press, Cambridge, New York, USA, 1995.

31. Holten, R. "Deriving an IS-Theory from an Epistemological Position," 18th Australasian Conference on Information Systems (ACIS), Toowoomba, Queensland, Australia, 2007, pp. 110-120.
32. Joshi, K.D., Sarker, S., and Sarker, S. "Knowledge transfer within information systems development teams: Examining the role of knowledge source attributes," *Decision Support Systems* (43:2) 2007, pp 322-335.
33. Kamlah, W., and Lorenzen, P. *Logische Propädeutik. Vorschule des vernünftigen Redens* Taschenbuch Verlag, Mannheim, Germany, 1967.
34. Kamlah, W., and Lorenzen, P. *Logical Propaedeutic. Pre-School of Reasonable Discourse* University Press of America, Lanham, MD, USA, 1984.
35. Ko, D.-G., Kirsch, L.J., and King, W.R. "Antecedents of Knowledge Transfer from Consultants to Clients in Enterprise System Implementations," *MIS Quarterly* (29:1) 2005, pp 59-85.
36. Langefors, B. *Essays on Infology* Studentlitteratur, Lund, Sweden, 1995.
37. Laumann, M. "Requirements Specification for Supply Chain Controlling - A Mini Case Study," 14th Americas Conference on Information Systems (AMCIS 2008), Toronto, Canada, 2008.
38. Laumann, M., and Rosenkranz, C. "Analysing Information Flows for Controlling Activities within Supply Chains - An arvato (Bertelsmann) Business Case," 16th European Conference on Information Systems (ECIS 2008), AIS, Galway, Ireland, 2008.
39. Lyytinen, K.J. "Implications of Theories of Language for Information Systems," *MIS Quarterly* (9:1) 1985, pp 61-76.
40. March, J.G., and Simon, H.A. *Organizations* Jon Wiley & Sons, New Yok, NY, USA, 1958.
41. Mintzberg, H. *The Nature of Managerial Work* Harper and Row, New York, NY, USA, 1973.
42. Moldoveanu, M.C. "Language, games and language games," *Journal of Socio-Economics*
43. (31:3) 2002, pp 233-251
44. Moody, D.L. "Theoretical and practical issues in evaluating the quality of conceptual models: current state and future directions," *Data & Knowledge Engineering* (55) 2005, pp 243-276.
45. Nikolopoulos, A., and Holten, R. "Analysis of E-Learning Implementation Cost Pools," Proceedings of the 18th Australasian Conference of Information Systems (ACIS 2007), Toowoomba, 2007, pp. 407-416.
46. Nordbotten, J.C., and Crosby, M.E. "The effect of graphic style on data model interpretation," *Information Systems Journal* (9:2) 1999, pp 139-155.
47. Pawlowski, S.D., Robey, D., and Raven, A. "Supporting shared information systems: boundary objects, communities, and brokering," Association for Information Systems Atlanta, GA, USA, 2000, pp. 329-338.
48. Radman, Z. *Metaphors: Figures of the Mind* Kluwer Academic Publishers, Dordrecht, The Netherlands et al., 1997.
49. Ribbers, P.M.A., and Schoo, K.-C. "Program Management and Complexity of ERP Implementations," *Engineering Management Journal* (14:2) 2002, pp 45-52.
50. Rosenkranz, C., Holten, R., and Laumann, M. "Designing IC Structures by Variety Engineering," 23nd Annual ACM Symposium on Applied Computing (SAC 2008), ACM, Fortaleza, Brazil, 2008, pp. 518-523.
51. Rosenkranz, C., Laumann, M., and Holten, R. "Diagnosing and Redesigning a Health(y) Organisation: An Action Research Study," *International Journal of Information Technologies and the Systems Approach* (2:1) 2009, pp 33-47.
52. Scheer, A.-W. *ARIS - Business Process Frameworks*, (3 ed.) Springer, Berlin et al., 1999.
53. SGI "(Standish Group International) Extreme CHAOS," Standish Group International, Inc, 2001.
54. Stachowiak, H. *Allgemeine Modelltheorie*, Wien, 1973.
55. Tan, M. "Establishing Mutual Understanding in Systems Design: An Empirical Study," *Journal of Management Information Systems* (10:4) 1994, pp 159-182.
56. Tushman, M.L. "Special Boundary Roles in the Innovation Process," *Administrative Science Quarterly* (22:4) 1977, pp 587-605.
57. Tushman, M.L. "Work Characteristics and Subunit Communication Structure: A Contingency Analysis," *Administrative Science Quarterly* (24:1) 1979, pp 82-98.
58. Tushman, M.L., and Nadler, D.A. "Information Processing as an Integrating Concept in Organizational Design," *The Academy of Management Review* (3:3) 1978, pp 613-624.
59. Vaast, E., and Walsham, G. "Representations and actions: the transformation of work practices with IT use," *Information and Organization* (15:1) 2005, pp 65-89.
60. Vlaar, P.W.L., Fenema, P.C.v., and Tiwari, V. "Cocreating Understanding and Value in Distributed Work: How Members of Onsite and Offshore Information Systems Development Vendor Teams Give, Make, Demand and Break Sense," *MIS Quarterly* (32:2) 2008, pp 227-255.
61. Wand, Y., and Weber, R. "On the ontological expressiveness of information systems analysis and design grammars," *Journal of Information Systems* (3:4) 1993, pp 217-237.

62. Wand, Y., and Weber, R. "On the Deep Structure of Information Systems," *Information Systems Journal* (5:3) 1995, pp 203-223.
63. Wand, Y., and Weber, R. "Research Commentary: Information Systems and Conceptual Modeling - A Research Agenda," *Information Systems Research* (13:4) 2002, pp 363-376.
64. Weber, R.A., and Camerer, C.F. "Cultural Conflict and Merger Failure: An Experimental Approach," *Management Science* (49:4) 2003, pp 400-415.
65. Weick, K.E. *The Social Psychology of Organizing*, (2 ed.) Addison Wesley, Reading, MA, USA, 1979.
66. Winograd, T. "A language/action perspective on the design of cooperative work," *Human-Computer Interaction* (3:1) 1988.
67. Winograd, T. "Designing a new foundation for design," *Communications of the ACM* (49:5) 2006, pp 71-74.