# Organizational Paradigms and Organizational Modelling

Joerg Evermann

School of Information Management, Victoria University, Wellington, New Zealand
Joerg.Evermann@mcs.vuw.ac.nz

**Abstract.** This paper introduces a new dimension to the alignment between business and information technology. Besides alignment of strategies, objectives, and processes, the paper argues that modelling languages for the description of organizational domains during requirements engineering must be aligned with the organizational paradigm. This paper examines five organizational paradigms and their ontologies. Current conceptual modelling languages are shown to be poorly aligned with the paradigms, pointing to a need for further research into adapting or creating suitable languages.

## 1 Introduction

Previous research on alignment of business and information technology or information systems has focused on strategy [1], objectives [2, 3], knowledge and competencies [4], and the IS function [5]. On an operational level, alignment has been discussed as the fit between business processes and the functionality of information systems [6, 7]. This paper describes a new type of alignment, the alignment between the organizational paradigm and modelling languages used for describing an organizational domain.

Alignment between the business and its information systems may be increased by information systems that reflect and support the business or organization (e.g. [6, 7]). Hence, it is necessary to first understand and describe the organization. Conceptual modelling, part of requirements engineering, is the activity of analyzing and describing organizational domains using formal or semi-formal modelling languages [8].

The elements of the organizational domain are determined by the perspective, point of view, or world-view of the organization, i.e. the organizational paradigm. This paradigm, in the Kuhnian sense, encompasses an ontology, a description of entities that exist in the organizational domain [9]. Different stakeholders may operate in different organizational paradigms.

In order to allow a complete description of the organizational domain, the conceptual modelling language must provide constructs for all elements of the organizational domain, as determined by the ontology of the relevant organizational paradigm. We define the *alignment between the conceptual modelling language and the organizational paradigm* as the extent to which a language provides constructs to describe the elements of the organizational ontology.

Such an alignment is important to enable and foster communication during IS development and IS usage. The importance of communication and communication skills has been recognized in the literature on IS project success [10–12] and user satisfaction [13], and has been explored in some detail in case studies [14]. However, the depth of analysis has been limited to assessing the effectiveness of the communication. As yet, there are no studies that analyze communication effectiveness at the level of the languages in use, and their alignment with the ontology held by different stakeholders. We argue in this paper that this alignment is an important factor in establishing effective communication: If the available language cannot convey that what is necessary to communicate, communication will be less effective.

In the remainder of this paper, we further discuss the notion of language-paradigm alignment (Section 2). We identify current organizational paradigms and their ontology, based on sociological work by Reed [15]. For each paradigm, we discuss alignment with UML and other example languages (Section 3). The paper concludes with an outlook to and call for further research (Section 4).

This paper should be understood as an initial, limited scope, effort. The main purpose of the paper is to highlight the existence of different organizational paradigms and their importance for business-IS alignment through conceptual modelling and language-paradigm alignment. UML and other languages are used as examples. The discussion remains informal; further research is clearly necessary to examine in detail the organizational paradigms and their implications for business-IS alignment.

## 2 Language-Paradigm Alignment

In section 1 we have defined language-paradigm alignment as the extent to which a language provides constructs for describing elements of the organization. Hence, alignment is closely related to semantics. The semantics of a language construct is its mapping to elements of the business or organizational domain [16]. Consequently, language constructs possess no implicit semantics.

Hence, when examining language-paradigm alignment we cannot assume existing semantics, unless mappings to the organizational domain are specified by the language creator. This is not the case for modelling languages such as ER diagrams or UML, when used for conceptual modeling.

However, most modellers assign *intuitive semantics* to language constructs. For example, when using UML, business analysts may map the 'object' construct to physically existing things. However, this need not be the case, and the 'object' construct may be mapped e.g. to software artifacts (its intended semantics), to events, processes, actors, etc. This flexibility of assigning different semantics to a language construct has been exploited in [17] to adapt modelling languages to a specific domain ontology, i.e. to align them with the domain.

The approach advocated here is to examine language-paradigm alignment in two steps. First, we establish a mapping between language constructs and organizational concepts, aided by intuitive semantics. For example, we may want to

represent human actors by the UML 'InstanceSpecifier' construct, rather than the 'Event' construct, because the name of the construct 'Event' evokes an intuitive semantics not compatible with our understanding of human actors[1].

If the first step is successful, the second step examines the syntactic restrictions between language constructs, and their implications for organizational concepts. For example, when representing actors by the 'InstanceSpecifier' and actors' goals by the 'Property' construct, UML suggests that actors may have zero or more goals, that the goals of each actor are ordered, that goals have an inverse goal, that goals may be composite or derived. These implications may or may not agree with the organizational ontology of the adopted organizational paradigm.

The large number of possible mappings between language constructs and organizational concepts, even when constrained by intuitive semantics, increases the difficulty in establishing high alignment when using complex languages such as UML. The alternative to this is the definition of new conceptual modelling languages with an explicitly defined semantics, based directly on a given organizational paradigm. We return to this discussion in Section 4.

## 3 Organizational Paradigms

Organizational paradigms and differences between them have been recognized as important for IS research. For example, [18, 19] discuss organizational paradigms in the context of IS development projects. However, the alignment of modelling languages with organizational paradigms has not been discussed.

The present research is based on work in organizational sociology by Reed [15]. Reed identifies five perspectives from which sociologists view organizations. Reed calls these perspectives frameworks, but explicitly recognizes them as paradigms in the Kuhnian sense, i.e. with their own distinct ontology. Discussions of these paradigms are also found in [20, 21]. Since different IS development stakeholders may operate in different paradigms [18, 19], it is important to model an organization from *all* perspectives to gain a complete understanding,

The following subsections provide a brief description of each paradigm and its ontology, based on the discussion in [15]. Ontological terminology taken from [15] is *emphasized*. Each subsection also discusses the relevance of the paradigm and its potential implications for IS development. It gives examples of language-paradigm alignment, using example languages such as UML.

Note again that the paper is intended to raise awareness of different organizational paradigms and their implications for conceptual modelling languages. The discussion in this section remains necessarily informal and incomplete.

### 3.1 Organizations as Social Systems

In this paradigm, *organizations* are composed of *social units*. These social units play *roles* in a *system*. The system has an *environment* that imposes *constraints*

[1] This paper uses UML 2.0 terminology.

on it. *Structural relationships* and *structural properties* of the system are designed to integrate the social units into a *coherent* and *stable* organization. The *internal structures* form networks of interrelated *sub-systems*. Organizations are characterized by recurring *properties* that define the roles and the organization. The *purpose* of the organization is to fulfill or satisfy environmentally derived *goals*, such as profit maximization in a capitalist environment. There exist *behavioural norms* with which the social units must *comply*. The main issue for organizations in this paradigm is to define the *structural design* that is appropriate to the fulfillment of *functional needs*. Examples of the systems paradigm are found in [22–24], the latter also being a critique of alternative paradigms.

This paradigm of organizations as systems of rationally interacting components corresponds well with the intuitive semantics of current conceptual modelling languages, such as UML. While there is no 'System' and 'Environment' construct in UML, social units correspond intuitively to 'InstanceSpecifications', i.e. objects which can be aggregated or composed. UML also provides 'Relationship' and 'Property' constructs, which can represent relationships and properties of the organizational units. Stability of systems can be represented in state machines, e.g. using 'State' and 'Trigger' constructs. The fact that social units may play roles can be represented either by classifying them using 'Class' constructs, or using the 'CollaborationRole' construct. Behavioural norms and other constraints may be represented by the 'Constraint' construct in UML. Needs and goals of social units may be represented by the 'Property' construct.

These brief examples show that an intuitive correspondence of UML to the ontology of this paradigm of organizations as systems can be found. In the second step, we need to determine whether the implications of this correspondence are compatible with the ontology of this paradigm. For example, representing sub-system relationships by the 'Association' construct implies that these relationships may have opposites, may be redefined, or may be derived. Similarly, representing roles by the 'CollaborationRole' construct implies that these roles are defined within a social unit that social units can play multiple roles. While a detailed analysis remains necessary to determine the validity of all such implications, current languages such as UML appear to be well-aligned with this paradigm.

### 3.2 Organizations as Negotiated Orders

While the systems perspective is likely the most prevalent, it is not the only one. When viewing organizations as negotiated orders, they are the *temporary product* of the *interactions* between *groups* or *individuals*. This product is temporary in the sense that it is continually re-created. An organization has no permanent systemic structure as in the previously discussed paradigm. The attention is on the *creation* and *transformation* through *social interactions*, an organization is the temporary *pattern* of such interactions. A few, very basic and general, structural conventions form the background and backbone against which the interactions occur. Organization does not exist separately from the constant *negotiations* between groups and individuals. There are no organizational goals

or functions, other than the *individual goals* and *individual functions* of the actors. It is the individual *interpretation* and *judgment* that determines the effects of any formal structures that may exist. Any power structures in an organization are also seen as temporary patterns in the negotiation process, arising out of conflicting *interests* and *values* of individuals. In this paradigm, the way to understand an organization is recognizing and analyzing the processes of negotiation and the patterns that arise out of them. A discussion of this paradigm is found in [25] and examples are presented in [26].

When examining UML for constructs with an intuitively matching semantics, we find that interactions can be expressed using 'Interaction', 'Message', 'Transition', 'Signal', and other related constructs. These appear to intuitively fit with the ontology of this paradigm. However, a 'Message' construct in UML implies existence of a 'Relationship' or 'Link', which is explicitly denied by this organizational paradigm. As another example, interpretation and judgment of individuals may be represented by the UML 'Property' construct. This however implies that interpretation and judgment may have opposites. The opposite of an interpretation or judgment is not an element of the domain. Other aspects, such as emergence or temporary existence is are also difficult to express in UML.

While the above may be reasonable intuitive mappings, with those mappings UML cannot differentiate between all ontological concepts. For example, interpretation and judgment are mapped to the same language construct. Furthermore, as the example of the 'Message' construct shows, the implications from the mapping are not always compatible with the ontology of this paradigm.

Neglecting the reality of agents, their interactions and goals can lead to information systems that fail to gain acceptance and that fail to support vital existing, but informal, interactions. This can lead to negative, disruptive effects, e.g. informal efficient organizations may be replaced by more formal but less efficient structures. For example, Yu et al. have shown benefits in the areas of work flow design [27, 28], the analysis of trust and vulnerabilities in system design [29], and patient care [30]. Their i* approach allows modelling individual agents with their rationales, dependencies. It allows describing social agents and their autonomy, intentionality, sociality, identity, self-interest, and reflectivity. Winograd & Flores [31] show how explicit consideration of actors and their interactions can benefit system design. Woo et al. [32, 33] recognizes the importance of analyzing and interactions between individual actors for workflow management systems.

### 3.3 Organizations as Structures of Power and Domination

In this paradigm, the only structures recognized as permanent and ontologically real are *power* and *power relationships*. These *constrain* and limit the *actions*, *activities*, and *interactions* of *social actors*. Additionally, organizations are *instruments* and *mechanisms* geared to the protection of *power structures* and *individual interests* of economic or political nature. Within these power structures and constraints, individuals aim for *control* of techniques and resources. *Conflicts* between *values* and *interests* of individuals produce tensions which are *regulated* by *managerial practices* to sustain *productivity*. These practices

are themselves subject to power and control struggles of individuals. At the center of the organization is a dynamic process of actions and interactions aimed at the control of *resources* and administrative and practices. Organizational structures are emergent, they are created and re-created by the dynamic processes. These structures are tools that serve interests of some actor. Examples of this paradigm are found in [34, 35].

In this paradigm, conceptual modelling languages and formalisms are required for describing power and control structures. For example, while it is possible to map UML 'InstanceSpecification' to power, and 'Relationship' to power structures, such an interpretation is relatively broad and does not support a deeper analysis of the organization. It would be difficult to describe the influence of actions on power structures, or the constraints that power structures impose on them. If, on the other hand, power structure were to be mapped to the UML 'constraint' construct, it would be difficult to describe dynamic and emergent power structures. Current languages also appear to fall short when describing notions such as values and interests and their relationships to power and control.

Failure to address power relationships and power structures in an organization may jeopardize completion of IS development and acceptance of the IS, as it can serve to threaten or undermine existing power positions. The IS may also be used as a tool to cement existing power structures, rather than to improve efficiency of the organization [36]. Power relationships are also relevant in the context of entire industries. The impact of an IS on industry structures has been demonstrated by Tillquist et al. [37] who explicitly analyze dependencies and power relationships. However, their proposed language, DND, does not include constructs for e.g. conflict, control, instruments, etc.

### 3.4 Organizations as Symbolic Constructions

This paradigm focuses on cultural and symbolic aspects, instead of structures and material components. This paradigm deals with the *myths*, *rituals*, *histories*, and *narratives* to be found in an organization. In this framework, organizations are constantly being created, reproduced, and changed through *symbolic construction*, *interpretation*, and *enactment* of *meanings*. Organizational culture is the product of the generation of *values*, *ideologies*, *rituals*, and *ceremonies* that make sense of the participation within the enterprise. Organizations are constructs of collective values and *symbols*. In this paradigm, the structural, technical and material components of an organization are continually interpreted and re-interpreted and assigned *shared meanings* as part of the process of organization. Organizational sub-cultures, offering *meaning* and *legitimation* for actions, are often in *conflict* with others, e.g. of different organizational units or departments. Examples of this framework are found in [38, 39].

The reality of organizations as symbolic constructions has been demonstrated by [18] and [19], who show the usefulness of metaphors in managing IS development projects. Working with existing symbolisms, rather than being ignorant or working against them, can increase development success. Failure to be aware

of the symbols, rituals, and ceremonies may cause systems to replace or restrict their availability and thus change the existing organizational structures in a way that is not intended or expected. Rituals and ceremonies contain meaning and purpose that must be understood in order to capture it in IS functionality. Rather than designing systems that hinder or restrict existing symbols, better user acceptance may be gained by systems that support the existing symbolism.

While work by [18, 19] show the relevance and managerial implications of this paradigm, it does not extend to the analysis of modelling languages. Current conceptual modelling languages appear poorly suited to describe concepts such as interpretation, construction and symbolism. While one could for example interpret UML 'InstanceSpecifier' (objects) as representing meaning, values, rituals, etc. this would preclude a finer differentiation, and result in a very generic description. For example, it would be difficult to describe which meanings, rituals, and ceremonies are related or contradict each other.

### 3.5 Organizations as Social Practices

In this paradigm, organizations are conceptualized as *social practices* that work towards the integration of other social practices. Ultimately, these practices serve to *transform* the material and ideal conditions under which social *action* takes place. Organizations are continually reproduced through the design and deployment of *administrative mechanisms* intended to regulate and control performance. Examples of such mechanisms are structural entities such as *hierarchies*, *rules*, *information systems*, etc. They require facilitating entities such as *ideologies* and *cultures* for their reproduction and implementation. Structural resources and the processual facilitators are subject to *power struggles* between competing groups and individuals. Organizations consist of administrative mechanisms, which in turn contain resources that enable productive activities. Social actors have *modes of reasoning*, *calculation*, and *deliberation* which directly determine their preferred outcomes and strategies to achieve them. This framework combines a number of important aspects of the previous four paradigms. It assigns ontological reality both to organizational structures as well as interpretation and symbolic meaning generating processes.

As this paradigm is an amalgamation of the ones discussed above, similar arguments apply to the reality of the paradigm, its importance and implications for IS development and the ability of conceptual modelling languages to capture its ontology. While the ontological reality of material and administrative structures lends itself to an application of current description languages such as UML, these languages offer little support for the remaining ontological elements in this paradigm, such as power structures and power struggles, the nature of social practices, and ideology and culture.

## 4 Discussion and Conclusion

The previous section illustrated the reality, importance and potential implications of different organizational paradigms for IS development and deployment.

We have argued that understanding and modelling the organization from different perspectives can offer valuable insights. While the systems paradigm is likely the prevailing one, different stakeholders view organizations from different perspectives. Even this brief examination of existing paradigms and languages shows that current conceptual languages appear ill-suited to describe organizational reality in all paradigms.

In summary, this paper has proposed a new dimension to business-IT alignment. This dimension of alignment is important to effective communication, a recognized factor in IS project success [10–12].

While the paper has used UML as an example, this should not be construed as an attempt to forcibly fit the UML language to these different ontologies. Instead, it shows the gap between the current de-facto description language and the ontologies that may be held by different stakeholders.

We argue with [15] that the organizational paradigms define different ontologies, rather than representing different levels of abstraction. For example, systems and systemic structures cannot easily be reconciled with notions of political power, culture, negotiation, and emergence. While IS analysts may not need to be able to integrate different stakeholders' paradigms, they must be able to "switch" and "think" in these different paradigms at different times, in order to communicate and understand requirements.

Successful IS development and deployment may be aided by the use of conceptual modelling languages with good alignment to these organizational ontologies. As the discussion in Sect. 2 has shown, there exist two ways of achieving this alignment. The first is to assign domain specific semantics to an existing language, based on the ontology of the organizational paradigm. This process is described in [17] and requires that the language and the ontology are already reasonably well aligned. The discussion in Sect. 3 has shown that this is the case for the example of UML and the systems paradigm. When mapping ontological concepts to language constructs, the implications that follow are in reasonable agreement with the ontology of the paradigm.

The second approach is to devise modelling languages that are directly based on the ontology of the paradigm. Work such as business model ontologies [40], goal modelling [41] and agent-oriented models [27–30, 42, 43] demonstrate this approach. However, more work is needed to explore the organizational paradigms and their ontological concepts.

## References

1. Sabherwal, R., Chan, Y.: Alignment between business and IS strategies. Information Systems Research **12** (2001) 11–33
2. Reich, B., Benbasat, I.: Measuring the linkage between business and information technology objectives. MIS Quarterly **20** (1996) 55–81
3. Reich, B., Benbasat, I.: Factors that influence the social dimension between business and information technoloy objectives. MIS Quarterly **24** (2000) 81–113
4. Bassellier, G., Benbasat, I., Reich, B.: The influence of business managers' IT competence on championing IT. Information Systems Research **14** (2003) 317–336

5. Brown, C., Magill, S.: Alignment of the IS function with the enterprise: Toward a model of antecedents. MIS Quarterly **18** (1994) 371–403
6. Etien, A., Roland, C., Salinesi, C.: Measuring the business / systems alignment. In: Proceedings of the 1st International Workshop of Requirements Engineering for Business Need and IT Alignment, REBNITA, Paris, France. (2005) 143–152
7. Kanyaru, J., Phalp, K.: Aligning business process models with specifications using enactable use case tools. In: Proceedings of the 1st International Workshop of Requirements Engineering for Business Need and IT Alignment, REBNITA, Paris, France. (2005) 22–31
8. Mylopoulos, J.: Conceptual modeling and Telos. In Locuopoulos, P., Zicari, R., eds.: Conceptual Modeling, Databases and Cases. John Wiley & Sons, Inc, New York et. al. (1992)
9. Kuhn, T.: The Structure of Scientific Revolutions. Third edn. The University of Chicago Press, Chicago (1996)
10. Group, S.: Extreme Chaos. (2001)
11. Keil, M., Tiwana, A., Bush, A.: Reconciling user and project manager perceptions of it project risk: a Delphi study. Information Systems Journal **12** (2002) 103–119
12. Schmidt, R., Lyytinen, K., Keil, M., Cule, P.: Identifying software project risks: An international Delphi study. Journal of Management Information Systems **17** (2001) 5–36
13. McKeen, J.D., Guimaraes, T., Wetherbe, J.C.: The relationship between user participation and user satisfaction. MIS Quarterly **18** (1994) 427–451
14. Hanish, J., Corbitt, B.: Requirements engineering during global software development: Some impediments to the requirements engineering process – a case study. In: Proceedings of the European Conference on Information Systems ECIS'04. (2004)
15. Reed, M.: The Sociology of Organisations: Themes, Perspectives and Prospects. Harvester Wheatsheaf, Hemel Hempstead, Hertfordshire (1992)
16. Harel, D., Rumpe, B.: Meaningful modeling: What's the semantics of "semantics"? IEEE Computer (2004) 64–72
17. Evermann, J., Wand, Y.: Towards formalizing domain modeling semantics in language syntax. IEEE Transactions on Software Engineering **31** (2005) 21–37
18. Kendall, J., Kendall, K.: Metaphors and methodologies: Living beyond the systems machine. MIS Quarterly **17** (1993) 149–172
19. Hirschheim, R., Newman, M.: Symbolism and information systems development: Myth, metaphor and magic. Information Systems Research **2** (1991) 29–62
20. Hatch, M.J.: Organization Theory - Modern, Symbolic, and Postmodern Perspectives. Oxford University Press, Oxford, UK (1997)
21. Thompson, P., McHugh, D.: Work Organizations - A Critical Introduction. 3rd ed. edn. Palgrave, New York, NY (2002)
22. Elliott, D.: The organization as a system. In Salaman, G., Thompson, K., eds.: Control and Ideology in Organizations. The Open University Press, Milton Keynes, UK (1980) 84–104
23. Thompson, J.: Organizations in Action. McGraw-Hill, New York, NY (1967)
24. Donaldson, L.: In Defence of Organizational Theory: A Reply to the Critics. Cambridge University Press, Cambridge, UK (1985)
25. Day, R., Day, J.V.: A review of the current state of negotiated order theory: An appreciation and a critique. The Sociological Quarterly **18** (1977) 126–142
26. Strauss, A.: Negotiations - Varieties, Contexts, Processes, and Social Order. Jossey-Bass Publishers, San Francisco, CA (1978)

27. Yu, E., Mylopoulos, J.: An actor dependency model of organizational work – with application to business process reengineering. In: Proceedings of COOCS'93. (1993) 258–268
28. Yu, E.: Models for supporting the redesign of organizational work. In: Proceedings of COOCS'95. (1995) 225–236
29. Yu, E., Liu, L.: Modelling trust for system design using the i* strategic actors framework. In Falcone, R., Singh, M., Tan, Y.H., eds.: Trust in Cyber-Societies. Springer-Verlag, Berlin (2001) 175–194
30. Yu, E.: Agent orientation as a modelling paradigm. Wirtschaftsinformatik **43** (2001) 123–132
31. Winograd, T., Flores, F.: Understanding Computers and Cognition: A New Foundation for Design. Addison Wesley, Reading, MA. (1986)
32. Janson, M.A., Woo, C.C., Smith, L.D.: Information systems development and communicative action theory. Information & Management **25** (1993) 59–72
33. Chang, M.K., Woo, C.C.: A speech-act-based negotiation protocol: Design, implementation, and test use. ACM Transactions on Information Systems **12** (1994) 360–382
34. Salaman, G.: Work Organizations - Resistance and Control. Longman, London, UK (1979)
35. Storey, J.: The means of management control. Sociology **19** (1985) 193–211
36. Hirschheim, R., Klein, H., Lyytinen, K.: Information Systems Development and Data Modeling: Conceptual and Philosophical Foundations. Cambridge University Press, Cambridge (1995)
37. Tillquist, J., King, J.L., Woo, C.: A representational scheme for analyzing information technology and organizational dependency. MIS Quarterly **26** (2002) 91–118
38. Gowler, D., Legge, K.: The meaning of management and the management of meaning: A view from social anthropology. In Earl, M.J., ed.: Perspectives on Management - A Multidisciplinary Analysis. Oxford University Press, Oxford, UK (1983) 197–233
39. Mangham, I.: Power and Performance in Organizations. Basil Blackwell Inc., Oxford, UK (1986)
40. Osterwalder, A.: The Business Model Ontology: A Proposition in a Design Science Approach. PhD thesis, Universite de Lausanne, Haute Ecole de Commerce (2004)
41. Lamsweerde, A.v.: Goal-oriented requriements engineering: A guided tour. In: Proceedings of the 5th IEEE International Symposium on Requirements Engineering. (2001) 249–262
42. Yu, E.: Agent-oriented modelling: Software versus the world. In Woolridge, M., Weiss, G., Ciancarini, P., eds.: Proceedings of the Workshop on Agent Oriented Software Engineering AOSE'01, Berlin, Springer-Verlag (2001) 206–225
43. Mao, X., Yu, E.: Organizational and social concepts in agent oriented software engineering. In: Proceedings of the Workshop on Agent Oriented Software Engineering AOSE'04, Berlin, Springer-Verlag (2004) 1–15