A Communicative Perspective on Second Order Information Systems

S.J.B.A. Hoppenbrouwers and H.A. Proper
Nijmegen Institute for Informatics and Information Science,
Radboud University Nijmegen,
Nijmegen, the Netherlands
stijnh@cs.kun.nl

Abstract
We discuss our preliminary efforts to create a generic theoretical model of „the socio-technical (information) systems that produce information systems (i.e. of *second order information systems*). We emphasize the importance of *communication, language, and meta-language* as factors in information system development processes and systems. Central are *conversations* related to the *specification* of information systems. Such conversations typically concern both formal and informal specification, and involve conceptual activities such as creation, adaptation, elicitation, informing, negotiation, validation, and committing. An integrated part of every specification process are conversations for *conceptualization*.

Keywords
Information system development, information systems, language, communication, modelling, specification, conversation.

Introduction

Much is known about technical and symbolical structures (often mathematics based) constituting the specifications and code of information and knowledge systems, but science and industry still fail to understand, and successfully govern, the complex chain of inter-dependent activities that is an information system development process. We consider *specification* (including *modelling*) to be a fundamental factor in such a process. High-level identification and prescription of strategies, phases, and key deliverables is commonplace in development theory and practice (e.g. DSDM Consortium, 2001; Kruchten, 2000), but more detailed, concept-oriented governance of the (iterative) process of conceiving, writing, and negotiating specifications is yet out of reach. Science and industry have so far mostly focused on the *products* of specification: all sorts of information system specifications in their final form. Initially, the focus lay on technological specifications (e.g. software architectures, program code), but technology-independent specifications (e.g. requirements, enterprise and process architectures, business rules) have recently been gaining importance in systems development. Many different stakeholders (users, managers, modellers, programmers, architects, etc.), with highly varied concerns (e.g. finance, operations, strategy, security, technology) may contribute to a specification process. Ultimately, many different sorts of specification (strategic, operational, technical, etc., *in various stages of maturity*, i.e. with highly different goals, sources, and contexts) should come together in an integrated and coherent *chain of development activities*. Yet currently there is a large gap between many such areas of specification, and in some the necessary specification efforts simply fail to take off. This seriously hampers system development. We believe that at least three key arguments can be put forward to strive for better *understanding* and *support* of specification processes. First, there is an increasing need for effective specification *in context*. Specification activities and deliverables should be dedicated to

the goals and stakeholders related to them them, which are rarely generic (the **effectivity** argument). Second, there is a rapidly increasing need for quick extension and adaptation of specifications because of factors like personalization and shortened time-to-market (the **flexibility** argument). Third, better efficiency and user friendliness of specification activities is required. There is a growing *specification bottleneck* related to increasing pervasiveness of IS systems in organizations and society. Automation and standardization can help here, but will not fully replace the need for well-conceived, highly contextualized specifications (Hoppenbrouwers, 2003, p32-3) (the **efficiency** argument). *Dynamics* and *rationales* with respect to effective, flexible, and efficient specification are still poorly understood. We therefore believe it is time to turn to detailed, scientific study of the whole of specification processes involved in IS development, complementary to and integrated with existing insights into what proper specifications should look like for particular areas in system development. A formidable knowledge creation, sharing, and negotiation effort underlies each sizable development project, and what makes all this possible is *communication*. We therefore advocate a view on system specification as a *rational communicative process* (Veldhuizen van Zanten et al., 2003). Its goal is to achieve good specifications that are an acceptable reflection of combined requirements and possibilities with respect to all stakeholders and concerns involved, so that stakeholders can all truly, realistically, and safely *commit to* and *support* system specifications as they are created, improved, and realized in system development.

Second Order Information Systems

Second order information systems (ISs) are information-and-communication systems that lead to the creation and adaptation of first order (operational) information-and-communication systems. Importantly, like first order ISs, second order ISs are not necessarily automated: they include all actors (both human and automated) involved in a development process, and the information-related activities these actors perform. In addition, there can be considerable overlap between the actors *developing* the system and the actors *using* the system (Hoppenbrouwers, 2003, p81). In this respect, second order information systems typically *envelop* their first order "offspring". Note that the existence of any first order information system implies the existence (in past or present) of a second order information system. Also note that the definition is recursive (it is possible to conceive third order and fourth order information systems, and so on). It is widely accepted that the future of information systems lies in Evolutionary Systems (Proper, 1994); the combination of a first and second order information system in fact represents a particular view on evolutionary information systems. Though second order information systems should ultimately also be seen as evolutionary, we propose to take a modest approach and first study them as static entities. The idea underlying this is that at a theoretical level, it is better feasible to study and describe stable structures that bring forth dynamic phenomena. In a later stage of research, principles found may indeed be recursively applied to 2nd order information systems, and so on.

Language and Communication in Second Order Information Systems

The explicit part of information systems (code, specifications, protocols, documentation) is essentially a collection of *texts*: coherent and meaningful conglomerates of statements. By means of creating texts, people shape their environment; communication builds on previously created texts *and also* creates new texts (Taylor, 1993). From a systemic point of view, texts are created and used by *actors* (in some research areas called agents or participants). Specifications (including

computer programs) then are essentially agreements and commitments holding between and interpreted by actors, rooted in social conventions (including language; Hoppenbrouwers, 2003; van Reijswoud, 1996). However, they may at the same time be part of actual computational machines shaping the physical environment. This gives rise to a particularly complex whole of concepts and texts, on both second order and first order level (and including interaction between the levels). Within systems theory, a particularly important link exists between second order information systems and Second Order Cybernetics (Von Foerster, 1995; Pask, 1975), a field which has been a main inspiration for our approach. Given the diversity of contexts, concerns, and goals involved, as reflected in a great diversity of actors (stakeholders) and also the concepts (languages, interpretations) that are wielded by and between them, the "whole of concepts and texts" referred to above very much hinges on carefully conducted and controlled *creation and sharing of meaning*. Consequently, conversation leading to effectively shared language and meaning is a fundamental and integral part of any attempt to develop an information system (Hoppenbrouwers, 2004). In line with the 2nd Order Cybernetic approach, we adopt a *constructivist* view on meaning and language. The concepts that people share and use are considered reflections of the contexts they live and work in, and this entails the need for *conceptual diversity*. Thus, properly conducted creation, negotiation, translation, specification, and sharing of elementary and complex concepts is (or should be) an essential part of all specification. Language is the linking pin between highly individual "thought" and persistent, explicit *structures* that shape the world --not just through communication, but also in creating and governing the computational machines and systems that physically shape our working environments, organisations, and societies.

A Sketch of a Generic Model of Second Order Information Systems

Our generic Specification Process Model (SPM) centres on *actors*, and the *communities* in which *conversations* among them take place. The model includes in principle the individual *speech acts* performed, and the *contexts* in which they take place, as well as *representations* (statements, texts) that may result from speech acts. Key properties of actors (communicative capacities; terms and languages they know; resources; concerns) and their *communication* and *conceptualisation* goals, are also to be captured in instantiated Specification Process analyses. Thus, it should be possible to perform a *rational analysis* of goals and means underlying specification conversations ·while making sure that sufficient respect is paid to the intricacies of communication and language. We aim to find and formulate detailed *strategies and tactics* used by actors in their specification conversations to fulfil the conversational goals by the means available to them. Thus, both *analysis* and *guidance* of specification comes within reach. Tactics for specification are anchored in creation, negotiation, and use of concepts in conversations and texts. Importantly, our rational analysis extends to the *utility of concepts* in view of IS development (Proper et al, 2004). Conversations both *about* concepts necessarily involve *meta-concepts*. Instead of assuming some standard set of meta-concepts, we include in our model the mechanisms for selecting, sharing, and adapting concepts and meta-concepts as required by the diverse contexts in the specification process (and activities relating them: negotiation, translation, refinement, etc.). Integrated in the specification process, therefore, are *conceptualisation* processes (Hoppenbrouwers, 2003). Many links can be observed between our approach and the field of *conceptual modelling*. Admittedly, our view on system development has been strongly influenced by work in conceptual modelling and domain modelling. Yet we emphatically put such modelling and specification in its wider IS development context, as described above. Importantly, we do not strive to capture specification and

conceptualisation processes in a strict step-by-step process model, and least of all a prescriptive one. Instead we wish to capture what drives and restrains actors in specification and conceptualisation activities, drawing on their natural language and communication abilities. However, we do want to add to humans' intuitive conceptualisation and specification capacities a goal-driven, functionalist and above all rational perspective on IS specification and development. Thus we hope to eventually combine the best of both worlds --intuitive "languaging" and rational specification--, and so realistically optimise specification processes (by making them more effective, flexible, and efficient).

Conclusion and Future Research

In this paper, we provided no more than a preliminary sketch of our model, and a basic rationale for creating it. The obvious next step (now in progress) is to put together a clear and concise formal "core model" that provides a good basis for further validation and refinement. Besides a generic model on a system-theoretical level, we will produce concrete specializations and even instantiations of the model concerning specific (and interrelated) fields of specification, for example enterprise architecture, requirements engineering, domain modelling, business process specification, programming, etc. We intend to validate and improve the SPM as sketched, using „field observations‰, as well as experimentational methods borrowed from 2nd order cybernetics; possibly Pask's Conversation Theory (Pask, 1975). Pask's work mostly concerns the teaching and learning of scientific topics (for example, probability theory), but the explicit (re)construction of mutually meaningful conceptual frameworks he focuses on is essentially quite close to what happens during any specification process. What is particularly interesting bout Pask's approach is that he combines a "microtheory" (descriptive and explanatory theory) with a "macrotheory" that provides means for the quantifying and measuring (in experimental setups) of activities covered by the microtheory. Finally a note about longer term applied directions. The SPM sketched above might eventually be used as a foundation for tools and frameworks supporting specification and conceptualisation processes, for example a *specification authoring environment*. Such an environment will integrate aspects of "decision support systems", "cooperative systems" (Computer Supported Cooperative Work), and "truth maintenance systems". It will obviously be both some sort of *knowledge based system* and a *knowledge management system*. What is special about it is that it will include the microdynamics of specification and modelling at the core of its underlying model.

References

DSDM Consortium (2001). Dynamic Systems Development Method, version 3.

H. von Foerster (1995). The Cybernetics of Cybernetics (2nd edition). Minneapolis: FutureSystems Inc.

S.J.B.A. Hoppenbrouwers (2003). Freezing Language: Conceptualisation Processes across ICT-Supported Organisations. PhD thesis, University of Nijmegen. http://webdoc.ubn.kun.nl/mono/h/hoppenbrouwers_s/freela.pdf

S.J.B.A. Hoppenbrouwers (2004). Towards Improved Mechanisms for Communication about Language in Information Systems Development. In: Proceedings of InterSymp 2003, Baden-Baden, Germany. Windsor, Ontario: IIAS.

P. Kruchten (2000). The Rational Unified Process: An Introduction. Reading, Massachusetts: Addison-Wesley. 2nd edition.

G. Pask (1975). Conversation, Cognition and Learning; a Cybernetic Theory and Methodology. Amsterdam: Elsevier Scientific Publishing.

H.A. Proper (1994); A Theory for Conceptual Modelling of Evolving Application Domains; PhD thesis, University of Nijmegen.

H.A. Proper, A.A. Verrijn-Stuart, and S.J.B.A. Hoppenbrouwers (1994). Utility-based Selection of Architecture-Modelling Concepts. Submitted to ER/Conceptual Modelling Conference, November 2004, Shanghai, China.

V.E. van Reijswoud (1996). The Structure of Business Communication: Theory, Model and Application. PhD thesis, Delft University of Technology.

J.R. Taylor (1993). Rethinking the Theory of Organizational Communication: how to Read an Organization. Norwood, New Jersey: Ablex Publishing.

G.E. Veldhuijzen van Zanten, S.J.B.A. Hoppenbrouwers, and H.A. Proper (2003). System development as a rational communicative process. In Proceedings of the 7th World Multiconference on Systemics, Cybernetics and Informatics, volume XVI, pages 126· 130, Orlando, Florida, USA, july 2003.